



UNIVERSITAS INDONESIA

**PERANCANGAN PENGENDALI *MODEL PREDICTIVE CONTROL (MPC) CONSTRAINED* PADA SISTEM
*PROTON EXCHANGE MEMBRANE FUEL CELL (PEM FC)***

TESIS

DHARMA ARYANI

0706173206

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KONTROL INDUSTRI
UNIVERSITAS INDONESIA
JULI 2009**



UNIVERSITAS INDONESIA

**PERANCANGAN PENGENDALI *MODEL PREDICTIVE CONTROL (MPC) CONSTRAINED* PADA SISTEM
*PROTON EXCHANGE MEMBRANE FUEL CELL (PEM FC)***

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister

DHARMA ARYANI

0706173206

**FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK KONTROL INDUSTRI
UNIVERSITAS INDONESIA
JULI 2009**

HALAMAN PERNYATAAN ORISINALITAS

**Tesis ini adalah hasil karya saya sendiri,
dan semua sumber yang baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar**

**Nama : Dharma Aryani
NPM : 0706173206
Tanda Tangan :**

Tanggal : Juli 2009

HALAMAN PENGESAHAN

Tesis ini diajukan oleh

Nama : Dharma Aryani
NPM : 0706173206
Program Studi : Teknik Kontrol Industri
Judul Tesis : Perancangan Pengendali *Model Predictive Control (MPC) constrained* pada sistem *Proton Exchange Membrane Fuel Cell (PEMFC)*

Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Teknik pada Program Teknik Elektro Fakultas Teknik Universitas Indonesia

DEWAN PENGUJI

Pembimbing I : Dr. Ir. Feri Yusivar, M.Eng. ()
Pembimbing II : Ir. Aries Subiantoro, M.SEE ()
Penguji : Dr. Ir. Ridwan Gunawan, MT ()
Penguji : Dr. Abdul Halim, M.Eng. ()

Depok, Juli 2009

UCAPAN TERIMAKASIH

Alhamdulillah, atas segala karunia dan petunjuk dari Allah SWT. Karena kasih-Nya sehingga penulis dapat menyelesaikan penyusunan laporan tesis ini. Penyusunan tesis ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Magister Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Untuk itu penulis mengucapkan terima kasih yang sangat mendalam kepada :

Bapak Dr. Ir. Feri Yusivar, M.Eng. dan Bapak Ir. Aries Subiantoro, M.SEE., selaku dosen pembimbing yang telah menyediakan waktu, tenaga dan pikiran didalam mengarahkan penulis selama penelitian dan penyusunan laporan tesis ini.

Dan semoga tesis ini membawa manfaat bagi pengembangan ilmu.

Depok, Juli 2009

Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Dharma Aryani

NPM : 0706173206

Program Studi : Teknik Kontrol Industri

Fakultas : Teknik

Jenis karya : Tesis

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Non- Eksklusif (*Non-exclusive Royalty-Free Right*)** atas karya ilmiah saya yang berjudul : Perancangan Pengendali *Model Predictive Control (MPC) constrained* pada sistem *Proton Exchange Membrane Fuel Cell (PEMFC)* beserta perangkat yang ada (bila diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/format-kan, mengelolanya dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok

Pada tanggal : Juli 2009

Yang menyatakan

(Dharma Aryani)

ABSTRAK

Nama : Dharma Aryani
Program Studi : Teknik Kontrol Industri
Judul Tesis : Perancangan dan *Model Predictive Control (MPC) Constrained* pada sistem *Proton Exchange Membrane Fuel Cell (PEMFC)*

Dalam thesis ini dirancang sebuah algoritma pengendali Model Predictive Control (MPC) Constrained dan diimplementasikan pada sistem Proton Exchange Membrane Fuel Cell. Model yang digunakan adalah model linier yang didapatkan dari Identifikasi sistem dengan metode Least Square. *Constraint* di berikan pada perubahan masing-masing sinyal kendali serta perbandingan antara sinyal kendali pertama dan kedua. Dari hasil simulasi terlihat bahwa pengendali MPC menghasilkan respon keluaran yang mengikuti sinyal acuan yang diberikan, serta mampu mengatasi gangguan yang berupa perubahan beban yang terjadi pada sistem PEMFC. Dengan pemberian *constraint* pada pengendali MPC, sinyal kendali yang dihasilkan dapat dibatasi sesuai dengan karakteristik fisik dari sistem PEMFC.

Kata kunci : MPC, Fuel Cell, Constraint, Model Linier

ABSTRACT

Name : Dharma Aryani
Study Program : Industrial Control Engineering
Thesis Title : Design of Model Predictive Control (MPC) Constrained for Proton Exchange Membrane Fuel Cell (PEMFC) system

This theses presents a Constrained Model Predictive Control design . The controller is implemented in the Proton Exchange Membrane Fuel Cell. The MPC algorithm based on the Linear model generated from identification system using Least Square Method. The controller consist of control signal constraints including the comparison of each control signal amplitude. The simulation result show that the MPC resulting a very good transient behaviour , the output from PEMFC can follow the trajectory and did not effected by load change disturbances. With some constraint additional in MPC, the control signals can be bounded refer to the real characteristic of PEMFC.

Keywords: MPC, Fuel Cell, Constrained, Linear Model

DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN JUDUL.....	ii
HALAMAN PERNYATAAN ORISINALITAS.....	iii
HALAMAN PENGESAHAN	iv
UCAPAN TERIMA KASIH	v
HALAMAN PERSETUJUAN PUBLIKASI	vi
ABSTRAK.....	vii
ABSTRACT.....	viii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Tujuan.....	2
1.3. Pembatasan Masalah.....	2
1.4. Susunan Penulisan.....	3
BAB 2 DASAR TEORI.....	4
2.1. Konsep Dasar MPC.....	4
2.2. Formulasi MPC.....	7
2.2.1. Model Proses.....	8
2.2.2. Prediksi	8
2.2.3. Strategi pengendali MPC Unconstrained.....	12
2.2.4. Strategi pengendali MPC Constrained	15
2.2.4.1. Pembentukan constrained	15
2.2.4.2. Metode Quadratic Programming.....	17
BAB 3 PERANCANGAN PENGENDALI MPC UNTUK SISTEM PEMFC.....	20
3.1. Sistem Dinamik PEMFC.....	20
3.2. Model Linier Sistem PEMFC.....	22
3.2.1. Pengetesan Observability dan Controllability Model.....	23

3.3 Algoritma MPC Constrained	25
3.4. Penurunan Persamaan Sinyal Kendali.....	30
3.5. Langkah Perhitungan Sinyal Kendali.....	31
BAB 4 ANALISA HASIL PERANCANGAN.....	37
4.1. Pengujian MPC Unconstrained.....	37
4.2. Pengujian MPC Constrained.....	38
BAB 5 KESIMPULAN.....	50
DAFTAR REFERENSI.....	51

DAFTAR GAMBAR

Gambar 2.1.	Struktur Pengendali MPC.....	6
Gambar 2.2.	Strategi Receding Horizon	6
Gambar 3.1	Karakteristik tegangan Fuel Cell	22
Gambar 3.2.	Blok Diagram Pengendali MPC dengan Constraint.....	25
Gambar 3.3.	Diagram alir algoritma MPC dengan constraint.....	26
Gambar 3.4.	Diagram alir metode active set untuk menyelesaikan QP...	29
Gambar 4.1.	Keluaran Sistem dengan pengendali MPC tanpa Constraint	37
Gambar 4.2.	Sinyal kendali pengendali MPC tanpa constraint.....	38
Gambar 4.3.	Keluaran Sistem dengan pengendali MPC Constraint(Q=1,R=0.1)	39
Gambar 4.4.	Sinyal kendali pengendali MPC constraint (Q=1,R=0.1) ..	39
Gambar 4.5.	Keluaran Sistem dengan pengendali MPC Constraint(Q=1,R=1)	40
Gambar 4.6.	Sinyal kendali pengendali MPC constraint (Q=1,R=1)	41
Gambar 4.7.	Keluaran Sistem dengan pengendali MPC Constraint(Q=1,R=10)	42
Gambar 4.8.	Sinyal kendali pengendali MPC constraint (Q=1,R=10) ..	42
Gambar 4.9.	Keluaran Sistem dengan pengendali MPC Constraint(Q=1,R=100)	43
Gambar 4.10.	Sinyal kendali pengendali MPC constraint (Q=1,R=100) ..	44
Gambar 4.11.	Keluaran Sistem dengan pengendali MPC Constraint(Q=0.1,R=10)	45
Gambar 4.12.	Sinyal kendali pengendali MPC constraint (Q=0.1,R=10) ..	46
Gambar 4.13.	Keluaran Sistem dengan pengendali MPC Constraint(Q=10,R=10)	47
Gambar 4.14.	Sinyal kendali pengendali MPC constraint (Q=10,R=10) ..	47
Gambar 4.15.	Keluaran Sistem dengan pengendali MPC Constraint(Q=100,R=10)	48
Gambar 4.16.	Sinyal kendali pengendali MPC constraint (Q=100,R=10)	49

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Selama ini, energi listrik di dunia banyak menggunakan sumber energi yang tidak terbarukan dengan cadangan yang semakin menipis. Hal ini mendorong dikembangkannya energi alternatif yang memiliki efisiensi tinggi dan ramah lingkungan. Dari sekian banyak alternatif energi, maka PEMFC (Proton Exchange Membrane Fuel Cell) menjadi salah satu solusi terbaik sebagai pembangkit listrik alternatif.

Pada fuel cell konversi energi terjadi dengan adanya reaksi elektrokimia dari bahan bakar (hidrogen) dan oksidan (oksigen) yang akan mengubah energi kimia fuel menjadi energi listrik. Tegangan keluaran yang dihasilkan oleh Fuel cell juga dipengaruhi oleh adanya rugi-rugi dari proses yang terjadi, baik dari proses kimia maupun mekanis. Karakteristik dari sistem fuel cell adalah tegangan yang dihasilkan akan semakin menurun jika terjadi penambahan beban. Sehingga dibutuhkan penambahan pengendali agar PEMFC dapat menghasilkan tegangan ideal.

Model Predictive Control merupakan suatu metodologi pengendalian yang saat ini memiliki pengaruh besar dalam dunia industri dibandingkan dengan pengendali konvensional seperti Two-Degree of Freedom ataupun Aturan Kendali Kenaikan.

Pada sistem kendali konvensional, batasan-batasan (constraints) seperti amplitudo dan slew rate sinyal kendali tidak diperhitungkan pada proses pengendalian. Hal ini tentu dapat menyebabkan hasil kendali menjadi kurang baik, terutama jika terjadi pemotongan paksa terhadap sinyal kendali sebelum masuk ke plant. Pemotongan sinyal kendali biasanya terjadi ketika nilai trayektori acuan berubah secara mendadak. Hal tersebut tentu tidak akan terjadi pada MPC karena pengendali dapat memprediksi keluaran proses yang akan datang serta tidak mengabaikan batasan-batasan yang ada. Selain agar keluaran sistem menjadi

bagus, adanya batasan pada proses pengendali dapat membuat kinerja alat menjadi optimal sehingga alat tidak cepat rusak dan dapat beroperasi dalam jangka waktu yang lama.

Banyaknya faktor yang harus diperhitungkan pada pengendali MPC membuat algoritma MPC menjadi sangat panjang dan rumit. Akan tetapi dengan kecepatan komputasi perangkat keras saat ini, tidak lagi menjadi masalah utama. Masalah utama metode MPC adalah keperluan akan model proses. Model proses pada MPC berguna untuk memprediksi keluaran sistem sehingga pengendali MPC dapat memberikan sinyal masukan yang sesuai. Oleh sebab itu, algoritma MPC membutuhkan model proses yang baik.

Model suatu sistem pada umumnya diperoleh melalui dua metode yaitu dengan pemodelan fisik dan identifikasi suatu sistem. Pemodelan fisik digunakan untuk mendapatkan model dengan menggunakan prinsip dasar persamaan differensial dan melakukan linierisasi pada suatu titik operasi. Sedangkan, identifikasi sistem adalah metode yang digunakan untuk mendapatkan model berdasarkan kumpulan data masukan dan keluaran hasil pengukuran sistem tersebut, dengan terlebih dahulu menetapkan struktur model sebelum parameter model ditentukan. Salah satu metode identifikasi yang paling banyak dipakai adalah metode kuadrat terkecil (Least Square).

1.2. Tujuan Penulisan

Pembahasan dalam thesis ini bertujuan untuk merancang sebuah pengendali Model Predictive Control dengan batasan-batasan, yang akan di aplikasikan pada Proton Exchange Membrane Fuel Cell (PEMFC) untuk menenteukan besar aliran Hydrogen dan Oxygen yang tepat dan mengendalikan tegangan keluaran fuel cell agar menghasilkan karakteristik ideal dari sistem PEMFC.

1.3. Pembatasan Masalah

Thesis ini membahas perancangan MPC dengan batasan, menggunakan metode *Quadratic Programming* dalam menghitung besar perubahan sinyal

kendali. Batasan (*Constraints*) yang digunakan adalah amplituda, *slew rate* sinyal kendali dan rasio aliran hydrogen terhadap oxygen.

Model linier yang digunakan dalam perancangan algoritma pengendali adalah model yang diperoleh dari hasil identifikasi Least Square, dengan aliran Hydrogen dan Oxygen sebagai masukan, perubahan kerapatan arus beban sebagai gangguan dan tegangan sebagai keluaran system. Parameter-parameter dan nilai fundamental dari system fisik PEMFC mengacu pada referensi utama yang digunakan.

1.4. Susunan Penulisan

Penulisan laporan thesis ini dibagi dalam lima bab yang akan menjelaskan secara bertahap mengenai keseluruhan isi skripsi ini.

Bab satu merupakan pendahuluan yang berisi latar belakang, tujuan, pembatasan masalah, dan sistematika penulisan. Bab dua membahas dasar teori yaitu tentang dasar teori Model Predictive Control dan model yang akan digunakan. Bab tiga membahas mengenai perancangan algoritma pengendali MPC. Bab empat hasil uji coba dan analisa terhadap hasil pengendalian yang dilakukan. Bab lima merupakan kesimpulan dari keseluruhan pembahasan dalam laporan thesis ini.

BAB 2

DASAR TEORI

2.1. Konsep Dasar *Model Predictive Control*

Model Predictive Control (MPC) atau sistem kendali prediktif termasuk dalam konsep perancangan pengendali berbasis model proses, dimana model proses digunakan secara eksplisit untuk merancang pengendali dengan cara meminimumkan suatu fungsi kriteria. Ide yang mendasari pada setiap jenis MPC adalah [2] :

1. Penggunaan model proses secara eksplisit untuk memprediksi keluaran proses yang akan datang dalam rentang waktu tertentu (*horizon*).
2. Perhitungan rangkaian sinyal kendali dengan meminimasi suatu fungsi kriteria.
3. Strategi surut; pada setiap waktu pencuplikan (pada waktu k) *horizon* dipindahkan menuju waktu pencuplikan berikutnya (pada waktu $k+1$) dengan melibatkan pemakaian sinyal kendali pertama (yaitu $u(k)$) untuk mengendalikan proses, dan kedua prosedur di atas diulang dengan menggunakan informasi terakhir.

Pada pengendali prediktif, model proses digunakan untuk memprediksi perilaku system dalam kurun waktu yang terbatas (*preceeding horizon*). Hasil prediksi ini digunakan untuk mengoptimasi keluaran system melalui sinyal masukan pada tiap waktu pencuplikan. Terdapat banyak variasi metode pengendali prediktif, dengan ide dasar yang sama yakni meminimasi fungsi criteria. Perbedaan metode-metode tersebut terdapat pada model proses, model derau, serta fungsi criteria yang diminimasi.

Metode MPC memiliki beberapa keunggulan dibandingkan dengan metode pengendali lainnya, di antaranya adalah :

1. Konsepnya sangat intuitif serta penalaannya mudah.
2. Dapat digunakan untuk mengendalikan proses yang beragam, mulai dari proses yang sederhana, hingga proses yang kompleks, memiliki

waktu tunda yang besar, *non-minimum phase* atau proses yang tidak stabil.

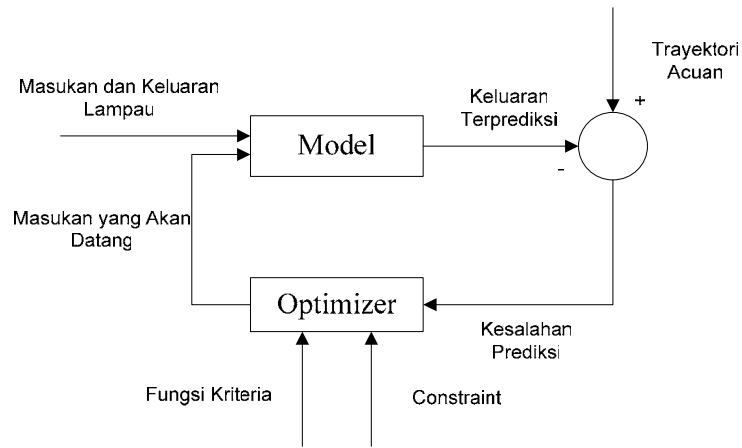
3. Dapat menangani sistem *multivariable*.
4. Mempunyai kompensasi terhadap waktu tunda.
5. Mempunyai kemampuan dari pengendali *feed forward* untuk mengkompensasi gangguan yang terukur.
6. Mudah untuk mengimplementasikan pengendali yang diperoleh.
7. Dapat memperhitungkan batasan atau *constraint* dalam merancang pengendali.
8. Sangat berguna jika sinyal acuan untuk masa yang akan datang diketahui.

Selain beragam keuntungan yang dimiliki, metode MPC juga mempunyai kelemahan, yaitu masalah penurunan aturan sinyal kendali yang cukup kompleks dan keperluan akan model proses yang baik. Struktur dasar dari pengendali MPC dapat dilihat pada gambar 2.1. Metodologi semua jenis pengendali yang termasuk kedalam kategori MPC dapat dikenali oleh strategi berikut [1] :

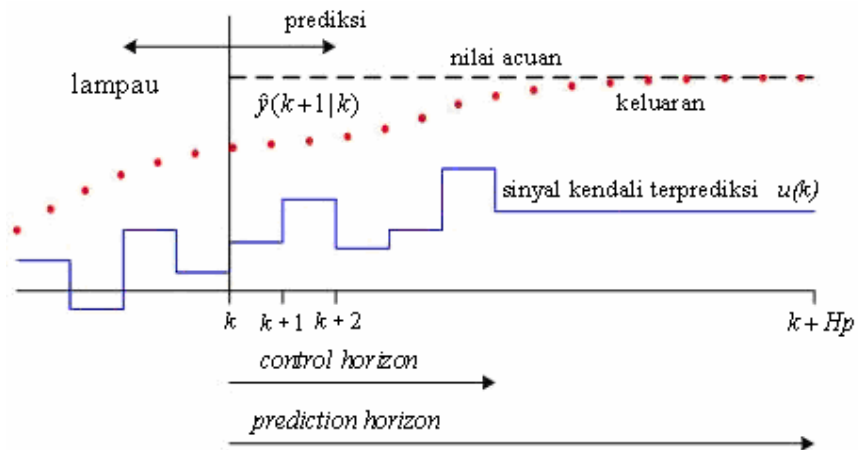
1. Keluaran proses yang akan datang untuk rentang *horizon Hp* yang ditentukan yang dinamakan sebagai *prediction horizon*, diprediksi pada setiap waktu pencuplikan dengan menggunakan model proses. Keluaran proses terprediksi ini $y(k+i|k)$ untuk $i = 1 \dots Hp$ bergantung pada nilai masukan dan keluaran lampau dan kepada sinyal kendali yang akan datang $u(k+i|k)$, $i = 0 \dots Hp-1$, yang akan digunakan sistem dan harus dihitung.
2. Serangkaian sinyal kendali dihitung dengan mengoptimasi suatu fungsi kriteria yang ditetapkan sebelumnya, dengan tujuan untuk menjaga proses sedekat mungkin terhadap trayektori acuan $r(k+i)$. Fungsi kriteria tersebut umumnya berupa suatu fungsi kuadratik dari kesalahan antara sinyal keluaran terprediksi dengan trayektori acuan. Solusi eksplisit dapat diperoleh jika fungsi kriteria adalah kuadratik, model linier, dan tidak ada *constraints*, jika tidak, optimasi iteratif harus

digunakan untuk memecahkannya. Langkah pertama dan kedua dapat diilustrasikan pada gambar 2.2.

3. Sinyal kendali $u(k|k)$ dikirim ke proses, sedangkan sinyal kendali terprediksi berikutnya dibuang, karena pada pencuplikan berikutnya $y(k+1)$ sudah diketahui nilainya. Maka langkah pertama diulang dengan nilai keluaran proses yang baru dan semua prosedur perhitungan yang diperlukan diperbaiki. Sinyal kendali yang baru $u(k+1|k+1)$ (nilainya berbeda dengan $u(k+1|k)$) dihitung dengan menggunakan konsep *receding horizon*.



Gambar2.1. Struktur pengendali MPC



Gambar2.2. Strategi Receding Horizon

2.2. Formulasi *Model Predictive Control*

Dalam formulasi dasar MPC ada beberapa asumsi yang dibuat yaitu, model acuan bersifat linier, fungsi kriteria merupakan fungsi kuadratik dan *constraints* dibentuk dalam pertidaksamaan dan persamaan linier. Seperti yang telah dinyatakan sebelumnya bahwa perhitungan sinyal kendali pada MPC dilakukan dengan meminimumkan suatu fungsi kriteria.

Fungsi kriteria yang digunakan dalam algoritma MPC berbentuk kuadratik seperti berikut

$$V(k) = \sum_{i=1}^{H_p} \|\underline{\hat{y}}(k+i|k) - \underline{r}(k+i|k)\|_{\underline{Q}(i)}^2 + \sum_{i=0}^{H_u-1} \|\underline{\Delta \hat{u}}(k+i|k)\|_{\underline{R}(i)}^2 \quad (2.1)$$

dengan :

$\underline{\hat{y}}(k+i|k)$ = keluaran terprediksi untuk i -langkah kedepan saat waktu k

$\underline{r}(k+i|k)$ = nilai trayektori acuan (*reference trajectory*)

$\underline{\Delta \hat{u}}(k+i|k)$ = perubahan nilai sinyal kendali terprediksi untuk i -langkah kedepan saat waktu k

$\underline{Q}(i)$ dan $\underline{R}(i)$ = faktor bobot

H_p = *prediction horizon*

H_u = *control horizon*

Dari persamaan fungsi kriteria tersebut, selalu dibuat asumsi bahwa nilai $H_u < H_p$ dan $\underline{\Delta \hat{u}}(k+i|k) = 0$ untuk $i \geq H_u$, sehingga nilai masukan terprediksi $\underline{\hat{u}}(k+i|k) = \underline{\hat{u}}(k+H_u-i|k)$ untuk semua $i \geq H_u$ seperti yang terlihat pada gambar 2.2.

Bentuk dari fungsi kriteria pada persamaan (2.1) menyatakan bahwa vektor kesalahan $\underline{\hat{y}}(k+i|k) - \underline{r}(k+i|k)$ dibebankan pada setiap rentang *prediction horizon*. Walaupun demikian tetap ada kemungkinan untuk menghitung vektor kesalahan pada titik-titik tertentu saja dengan cara mengatur matriks faktor bobot $\underline{Q}(i)$ bernilai nol pada langkah yang diinginkan. Selain vektor kesalahan, fungsi kriteria pada persamaan (2.1)

juga memperhitungkan perubahan vektor masukan dalam rentang *control horizon*. Pemilihan penggunaan $\underline{\hat{u}}(k+i|k)$ yang pada fungsi kriteria bertujuan untuk meminimumkan perubahan sinyal kendali yang masuk ke *plant*.

2.2.1. Model Proses

Pada perancangan MPC, umumnya model proses yang digunakan berupa model ruang keadaan diskrit linier seperti berikut :

$$\underline{x}(k+1) = \underline{A}\underline{x}(k) + \underline{B}\underline{u}(k) \quad (2.2)$$

$$\underline{y}(k) = \underline{C}\underline{x}(k) \quad (2.3)$$

dengan :

$\underline{u}(k)$ = vektor masukan berdimensi- l

$\underline{x}(k)$ = vektor keadaan berdimensi- n

$\underline{y}(k)$ = vektor keluaran berdimensi- m

\underline{A} = matriks keadaan berdimensi $n \times n$

\underline{B} = matriks masukan berdimensi $n \times l$

\underline{C} = matriks keluaran berdimensi $m \times n$

Persamaan ruang keadaan ini merupakan kondisi ideal dan sederhana untuk sebuah sistem, karena tidak adanya *disturbance* serta *directfeedthrough* pada keluaran sistem. Pada perancangan MPC di Bab selanjutnya, akan diuraikan penurunan sinyal kendali untuk model yang lebih kompleks yang digunakan.

2.2.2. Prediksi

Sinyal masukan yang digunakan dalam perhitungan prediksi keluaran adalah perubahan nilai sinyal masukan $\Delta u(k)$ pada setiap waktu pencuplikan k . Dimana perubahan tersebut merupakan selisih antara nilai sinyal masukan saat k atau $\underline{u}(k)$ dan sinyal masukan satu langkah

sebelumnya $\underline{u}(k-1)$. Dalam menyelesaikan masalah pengendali prediktif, nilai keluaran terprediksi $\hat{y}(k+i|k)$ harus dapat dihitung dengan menggunakan estimasi terbaik dari variabel keadaan saat ini $\underline{x}(k)$, nilai masukan yang lampau $\underline{u}(k-1)$, dan nilai perkiraan dari perubahan masukan yang akan datang $\underline{\Delta\hat{u}}(k+i|k)$. Sebelum melangkah lebih jauh, hal pertama yang harus dilakukan adalah memprediksi nilai variabel keadaan dengan melakukan iterasi model ruang keadaan pada persamaan (2.2) dan (2.3). Perhitungan prediksi variabel keadaan adalah sebagai berikut;

$$\hat{\underline{x}}(k+1|k) = \underline{A}\underline{x}(k) + \underline{B}\hat{\underline{u}}(k|k) \quad (2.4)$$

$$\begin{aligned} \hat{\underline{x}}(k+2|k) &= \underline{A}\hat{\underline{x}}(k+1|k) + \underline{B}\hat{\underline{u}}(k+1|k) \\ &= \underline{A}^2\underline{x}(k) + \underline{A}\underline{B}\hat{\underline{u}}(k|k) + \underline{B}\hat{\underline{u}}(k+1|k) \end{aligned} \quad (2.5)$$

⋮

$$\begin{aligned} \hat{\underline{x}}(k+Hp|k) &= \underline{A}\hat{\underline{x}}(k+Hp-1|k) + \underline{B}\hat{\underline{u}}(k+Hp-1|k) \\ &= \underline{A}^{Hp}\underline{x}(k) + \underline{A}^{Hp-1}\underline{B}\hat{\underline{u}}(k|k) + \dots + \underline{B}\hat{\underline{u}}(k+Hp-1|k) \end{aligned} \quad (2.6)$$

Pada setiap langkah prediksi digunakan $\hat{\underline{u}}(k|k)$ bukan $\underline{u}(k)$, karena besarnya nilai $\underline{u}(k)$ belum diketahui ketika menghitung prediksi.

Sekarang, diasumsikan bahwa nilai masukan hanya berubah pada waktu $k, k+1, \dots, k+Hu-1$, dan setelah itu menjadi konstan, sehingga didapatkan bahwa $\hat{\underline{u}}(k+i|k) = \hat{\underline{u}}(k+Hu-1|k)$ untuk $Hu \leq i \leq Hp-1$.

Selanjutnya, perhitungan prediksi diubah sehingga mengandung $\underline{\Delta\hat{u}}(k+i|k)$ daripada $\hat{\underline{u}}(k+i|k)$, dengan

$$\underline{\Delta\hat{u}}(k+i|k) = \hat{\underline{u}}(k+i|k) - \hat{\underline{u}}(k+i-1|k) \quad (2.7)$$

dan pada setiap waktu pencuplikan k nilai yang sudah diketahui hanya $\underline{u}(k-1)$, maka

$$\underline{\hat{u}}(k | k) = \underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1 | k) \quad (2.8)$$

$$\underline{\hat{u}}(k+1 | k) = \underline{\Delta\hat{u}}(k+1 | k) + \underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1 | k) \quad (2.9)$$

⋮

$$\underline{\hat{u}}(k+Hu-1 | k) = \underline{\Delta\hat{u}}(k+Hu-1 | k) + \dots + \underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1 | k) \quad (2.10)$$

Dengan mensubstitusikan persamaan (2.8) – (2.10) ke persamaan (2.4) – (2.6), diperoleh persamaan

$$\underline{\hat{x}}(k+1 | k) = \underline{A}\underline{x}(k) + \underline{B}[\underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1)] \quad (2.11)$$

$$\begin{aligned} \underline{\hat{x}}(k+2 | k) &= \underline{A}^2\underline{x}(k) + \underline{A}\underline{B}[\underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1)] \\ &\quad + \underline{B}[\underbrace{\underline{\Delta\hat{u}}(k+1 | k) + \underline{\Delta\hat{u}}(k | k) + \underline{u}(k-1)}_{\underline{\hat{u}}(k+1|k)}] \\ &= \underline{A}^2\underline{x}(k) + (\underline{A} + \underline{I})\underline{B}\underline{\Delta\hat{u}}(k | k) + \underline{B}\underline{\Delta\hat{u}}(k+1 | k) + (\underline{A} + \underline{I})\underline{B}\underline{u}(k-1) \end{aligned} \quad (2.12)$$

⋮

$$\begin{aligned} \underline{\hat{x}}(k+Hu | k) &= \underline{A}^{Hu}\underline{x}(k) + (\underline{A}^{Hu-1} + \dots + \underline{A} + \underline{I})\underline{B}\underline{\Delta\hat{u}}(k | k) + \dots \\ &\quad + \underline{B}\underline{\Delta\hat{u}}(k+Hu-1 | k) + (\underline{A}^{Hu-1} + \dots + \underline{A} + \underline{I})\underline{B}\underline{u}(k-1) \end{aligned} \quad (2.13)$$

Dengan mengacu pada persamaan $\underline{\hat{u}}(k+i | k) = \underline{\hat{u}}(k+Hu-i | k)$ untuk $i > Hu$, maka perhitungan prediksi untuk $i > Hu$ adalah;

$$\begin{aligned} \underline{\hat{x}}(k+Hu+1 | k) &= \underline{A}^{Hu+1}\underline{x}(k) + (\underline{A}^{Hu} + \dots + \underline{A} + \underline{I})\underline{B}\underline{\Delta\hat{u}}(k | k) + \dots \\ &\quad + (\underline{A} + \underline{I})\underline{B}\underline{\Delta\hat{u}}(k+Hu-1 | k) + (\underline{A}^{Hu} + \dots + \underline{A} + \underline{I})\underline{B}\underline{u}(k-1) \end{aligned} \quad (2.14)$$

⋮

$$\begin{aligned}
\hat{\underline{x}}(k + Hp | k) &= \underline{A}^{Hp} \underline{x}(k) + (\underline{A}^{Hp-1} + \dots + \underline{A} + \underline{I}) \underline{B} \underline{\Delta \hat{u}}(k | k) + \dots \\
&+ (\underline{A}^{Hp-Hu} + \dots + \underline{A} + \underline{I}) \underline{B} \underline{\Delta \hat{u}}(k + Hu - 1 | k) \\
&+ (\underline{A}^{Hp-1} + \dots + \underline{A} + \underline{I}) \underline{B} \underline{u}(k - 1)
\end{aligned} \tag{2.15}$$

Persamaan (2.11) – (2.15) dapat disusun ke dalam bentuk vektor matriks sebagai berikut

$$\begin{aligned}
\begin{bmatrix} \hat{\underline{x}}(k+1 | k) \\ \vdots \\ \hat{\underline{x}}(k+Hu | k) \\ \hat{\underline{x}}(k+Hu+1 | k) \\ \vdots \\ \hat{\underline{x}}(k+Hp | k) \end{bmatrix} &= \underbrace{\begin{bmatrix} \underline{A} \\ \vdots \\ \underline{A}^{Hu} \\ \underline{A}^{Hu+1} \\ \vdots \\ \underline{A}^{Hp} \end{bmatrix}}_{\Psi} \underline{x}(k) + \underbrace{\begin{bmatrix} \underline{B} \\ \vdots \\ \sum_{i=0}^{Hu-1} \underline{A}^i \underline{B} \\ \sum_{i=0}^{Hu} \underline{A}^i \underline{B} \\ \vdots \\ \sum_{i=0}^{Hp-1} \underline{A}^i \underline{B} \end{bmatrix}}_{\Gamma} \underline{u}(k-1) \\
&+ \underbrace{\begin{bmatrix} \underline{B} & \dots & \underline{0}_{n \times l} \\ \underline{AB} + \underline{B} & \dots & \underline{0}_{n \times l} \\ \vdots & \ddots & \vdots \\ \sum_{i=0}^{Hu-1} \underline{A}^i \underline{B} & \dots & \underline{B} \\ \sum_{i=0}^{Hu} \underline{A}^i \underline{B} & \dots & \underline{AB} + \underline{B} \\ \vdots & \vdots & \vdots \\ \sum_{i=0}^{Hp-1} \underline{A}^i \underline{B} & \dots & \sum_{i=0}^{Hp-Hu} \underline{A}^i \underline{B} \end{bmatrix}}_{\Theta} \begin{bmatrix} \underline{\Delta \hat{u}}(k) \\ \vdots \\ \underline{\Delta \hat{u}}(k+Hu-1) \end{bmatrix}
\end{aligned} \tag{2.16}$$

Selain itu, persamaan prediksi keluaran $\hat{\underline{y}}(k+i | k)$ dapat ditulis seperti berikut ini;

$$\hat{\underline{y}}(k+1 | k) = \underline{C} \hat{\underline{x}}(k+1 | k) \tag{2.17}$$

$$\hat{\underline{y}}(k+2 | k) = \underline{C} \hat{\underline{x}}(k+2 | k) \tag{2.18}$$

⋮

$$\hat{\underline{y}}(k+Hp | k) = \underline{C} \hat{\underline{x}}(k+Hp | k) \tag{2.19}$$

Persamaan (2.17) – (2.19) kemudian dapat ditulis kedalam vektor matriks sebagai berikut

$$\begin{bmatrix} \underline{\hat{y}}(k+1|k) \\ \vdots \\ \underline{\hat{y}}(k+Hp|k) \end{bmatrix} = \underbrace{\begin{bmatrix} \underline{C} & \underline{0}_{m \times n} & \cdots & \underline{0}_{m \times n} \\ \underline{0}_{m \times n} & \underline{C} & \cdots & \underline{0}_{m \times n} \\ \vdots & \vdots & \ddots & \vdots \\ \underline{0}_{m \times n} & \underline{0}_{m \times n} & \cdots & \underline{C} \end{bmatrix}}_{\underline{C}_y} \begin{bmatrix} \underline{\hat{x}}(k+1|k) \\ \vdots \\ \underline{\hat{x}}(k+Hp|k) \end{bmatrix} \quad (2.20)$$

2.2.3. Strategi Pengendali *Model Predictive Control* tanpa *Constraints*

Fungsi kriteria yang akan diminimumkan adalah fungsi kuadratik seperti pada persamaan (2.1) dan dapat ditulis sebagai berikut

$$V(k) = \|\underline{Y}(k) - \underline{T}(k)\|_{\underline{Q}}^2 + \|\underline{\Delta U}(k)\|_{\underline{R}}^2 \quad (2.21)$$

dimana

$V(k)$ = fungsi kriteria

$\underline{Y}(k)$ = matriks keluaran terprediksi

$\underline{T}(k)$ = matriks sinyal acuan (trajectory)

$\underline{\Delta U}(k)$ = perubahan sinyal kendali

$$\underline{Y}(k) = \begin{bmatrix} \underline{\hat{y}}(k+1|k) \\ \vdots \\ \underline{\hat{y}}(k+Hp|k) \end{bmatrix}, \quad \underline{T}(k) = \begin{bmatrix} \underline{r}(k+1|k) \\ \vdots \\ \underline{r}(k+Hp|k) \end{bmatrix},$$

$$\underline{\Delta U}(k) = \begin{bmatrix} \underline{\hat{u}}(k|k) \\ \vdots \\ \underline{\hat{u}}(k+Hu-1|k) \end{bmatrix}$$

dan matriks faktor bobot \underline{Q} dan \underline{R} adalah sebagai berikut

$$\underline{Q} = \begin{bmatrix} Q(1) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & Q(Hp) \end{bmatrix} \quad (2.22)$$

$$\underline{R} = \begin{bmatrix} R(0) & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & R(Hu-1) \end{bmatrix} \quad (2.23)$$

Bentuk fungsi kriteria menunjukkan bahwa vektor kesalahan (error) $\underline{Y}(k) - \underline{T}(k)$ diperhitungkan pada tiap pencuplikan dalam rentang *prediction horizon*, namun jika perhitungan error hanya diinginkan pada rentang waktu tertentu, hal ini dapat dilakukan dengan mengatur nilai faktor bobot Q bernilai 0 pada waktu tersebut. Selain vektor kesalahan, fungsi kriteria juga memperhitungkan perubahan dari vektor masukan yang hanya terjadi dalam rentang waktu *control horizon*.

Berdasarkan pada persamaan ruang keadaan (2.16) dan (2.20), maka matriks $\underline{Y}(k)$ dapat ditulis dalam bentuk;

$$\underline{Y}(k) = \underline{C}_y \underline{\Psi} \underline{x}(k) + \underline{C}_y \underline{\Gamma} \underline{u}(k-1) + \underline{C}_y \underline{\Theta} \underline{\Delta U}(k) \quad (2.24)$$

Selain matriks-matriks di atas, didefinisikan juga suatu matriks penjejukan kesalahan $\underline{\mathcal{E}}(k)$, yaitu selisih antara nilai trayektori acuan yang akan datang dengan tanggapan bebas dari sistem. Tanggapan bebas adalah tanggapan yang akan terjadi pada rentang *prediction horizon* jika tidak ada perubahan nilai masukan ($\underline{\Delta U}(k) = \underline{0}$) [3]. Persamaan matematis dari matriks $\underline{\mathcal{E}}(k)$ adalah sebagai berikut ;

$$\underline{\mathcal{E}}(k) = \underline{T}(k) - \underline{C}_y \underline{\Psi} \underline{x}(k) - \underline{C}_y \underline{\Gamma} \underline{u}(k-1) \quad (2.25)$$

Persamaan (2.21) kemudian dapat ditulis kembali dalam bentuk yang mengandung matriks $\underline{\mathcal{E}}(k)$ dan $\underline{\Delta U}(k)$ sebagai berikut ;

$$\begin{aligned}
V(k) &= \|\underline{C}_y \underline{\Theta} \underline{\Delta U}(k) - \underline{E}(k)\|_{\underline{Q}}^2 + \|\underline{\Delta U}(k)\|_{\underline{R}}^2 \\
&= \underline{\Delta U}^T(k) \underline{\Theta}^T \underline{C}_y^T - \underline{E}^T(k) \underline{Q} [\underline{C}_y \underline{\Theta} \underline{\Delta U}(k) - \underline{E}(k)] + \underline{\Delta U}^T(k) \underline{R} \underline{\Delta U}(k) \\
&= \underbrace{\underline{E}^T(k) \underline{Q} \underline{E}(k)}_{\underline{c}_1} - \underline{\Delta U}^T(k) \underbrace{2 \underline{\Theta}^T \underline{C}_y^T \underline{Q} \underline{E}(k)}_{\underline{G}} + \underline{\Delta U}^T(k) \underbrace{[\underline{\Theta}^T \underline{C}_y^T \underline{Q} \underline{C}_y \underline{\Theta} + \underline{R}]}_{\underline{H}} \underline{\Delta U}(k)
\end{aligned} \tag{2.26}$$

Pada persamaan (2.26), bagian $\underline{E}^T(k) \underline{Q} \underline{E}(k)$ tidak mengandung unsur $\underline{\Delta U}(k)$ sehingga bagian tersebut bisa dianggap konstan sehingga bagian tersebut tidak diikutsertakan dalam proses optimasi untuk menghitung nilai $\underline{\Delta U}(k)$. Persamaan (2.26) kemudian dapat ditulis kembali menjadi

$$V(k) = \underline{c}_1 - \underline{\Delta U}^T(k) \underline{G} + \underline{\Delta U}^T(k) \underline{H} \underline{\Delta U}(k) \tag{2.27}$$

dimana

$$\underline{G} = 2 \underline{\Theta}^T \underline{C}_y^T \underline{Q} \underline{E}(k) \tag{2.28}$$

dan

$$\underline{H} = \underline{\Theta}^T \underline{C}_y^T \underline{Q} \underline{C}_y \underline{\Theta} + \underline{R} \tag{2.29}$$

Nilai optimal $\underline{\Delta U}(k)$ dapat dihitung dengan membuat gradien dari $V(k)$ bernilai nol [3]. Gradien $V(k)$ dari persamaan (2.27) adalah

$$\nabla_{\underline{\Delta U}(k)} \underline{V}(k) = -\underline{\mathcal{G}} + 2\underline{\mathcal{H}} \underline{\Delta U}(k) \quad (2.30)$$

Dengan membuat nol nilai $\nabla_{\underline{\Delta U}(k)} \underline{V}(k)$ pada persamaan (2.30), maka didapatkan nilai optimal dari perubahan sinyal kendali sebagai berikut

$$\underline{\Delta U}(k)_{opt} = \frac{1}{2} \underline{\mathcal{H}}^{-1} \underline{\mathcal{G}} \quad (2.31)$$

Setelah nilai matriks $\underline{\Delta U}(k)$ didapatkan, maka nilai yang digunakan untuk mengubah sinyal kendali hanya nilai dari baris pertama matriks $\underline{\Delta U}(k)$ sedangkan nilai dari baris yang lain dari matriks $\underline{\Delta U}(k)$ dibuang [3].

2.2.4. Strategi Pengendali *Model Predictive Control* dengan *Constraints*

2.2.4.1. Pembentukan *Constraints*

Pada setiap kendali proses, pasti terdapat batasan atau *constraints* pada amplitudo sinyal kendali. Selain itu, besarnya *slew rate* sinyal kendali juga dapat menjadi batasan. Persamaan *constraints* untuk amplitudo dan *slew rate* sinyal kendali secara berturut-turut adalah sebagai berikut

$$\underline{FU}(k) \leq \underline{f} \quad (2.32)$$

$$\underline{E}\underline{\Delta U}(k) \leq \underline{e} \quad (2.33)$$

Pada algoritma MPC, yang akan dihitung adalah nilai optimal perubahan sinyal kendali $\underline{\Delta U}(k)$ sehingga sangat perlu untuk mengubah bentuk *constraints* yang belum mengandung $\underline{\Delta U}(k)$ menjadi bentuk *constraints* yang mengandung $\underline{\Delta U}(k)$. Sebagai contoh adalah pertidaksamaan (2.32), karena pada pertidaksamaan (2.32) belum mengandung $\underline{\Delta U}(k)$ maka bentuk pertidaksamaan (2.32) harus diubah terlebih dahulu menjadi bentuk yang mengandung $\underline{\Delta U}(k)$.

Untuk *constraints* yang berupa batasan nilai maksimum dan minimum sinyal kendali, maka pertidaksamaannya dapat ditulis sebagai berikut

$$\underline{u}_{\min} \leq \underline{u}(k) \leq \underline{u}_{\max} \quad (2.34)$$

Pertidaksamaan (2.34) dapat ditulis menjadi dua bentuk yang terpisah seperti berikut ini

$$-\underline{u}(k) \leq -\underline{u}_{\min} \quad (2.35)$$

$$\underline{u}(k) \leq \underline{u}_{\max} \quad (2.36)$$

Pertidaksamaan (2.35) dan (2.36) masing-masing dapat ditulis dalam bentuk yang mengandung $\underline{\Delta U}(k)$ menjadi

$$-\underline{F}'\underline{\Delta U}(k) \leq -\underline{u}_{\min} + \underline{F}_1 u(k-1) \quad (2.37)$$

$$\underline{F}'\underline{\Delta U}(k) \leq \underline{u}_{\max} - \underline{F}_1 u(k-1) \quad (2.38)$$

dimana

$$\underline{F}' = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 1 & 1 & 0 & \cdots & 0 \\ 1 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & 1 & \cdots & 1 \end{bmatrix}_{H_u \times H_u} \quad (2.39)$$

dan

$$\underline{F}_1 = \begin{bmatrix} 1 \\ \vdots \\ 1 \end{bmatrix}_{H_u \times 1} \quad (2.40)$$

Untuk pertidaksamaan (2.33), bentuknya tidak perlu diubah lagi karena pada pertidaksamaan tersebut sudah mengandung unsur $\underline{\Delta U}(k)$.

Pertidaksamaan (2.33), (2.37), dan (2.38) kemudian dapat disusun menjadi sebuah vektor matriks sebagai berikut

$$\underbrace{\begin{bmatrix} -\underline{F}' \\ \underline{F}' \\ \underline{E} \end{bmatrix}}_{\underline{\Omega}} \underbrace{\Delta U(k)}_{\underline{\delta}} \leq \underbrace{\begin{bmatrix} -\underline{u}_{\min} + \underline{F}_1 u(k-1) \\ \underline{u}_{\max} - \underline{F}_1 u(k-1) \\ \underline{e} \end{bmatrix}}_{\underline{\omega}} \quad (2.41)$$

Vektor matriks pada pertidaksamaan (2.41) digunakan pada perhitungan nilai optimal perubahan sinyal kendali $\underline{\Delta U}(k)_{opt}$.

2.2.4.2. Metode *Quadratic Programming*

Fungsi kriteria pada pengendali MPC dengan *constraints* sama dengan fungsi kriteria pada pengendali MPC tanpa *constraints* (persamaan (2.35)). Permasalahan utama proses optimasi ini adalah meminimalkan fungsi kriteria

$$\underline{\Delta U}^T(k) \underline{\mathcal{H}} \underline{\Delta U}(k) - \underline{\Delta U}^T(k) \underline{\mathcal{G}} \quad (2.42)$$

berdasarkan pada pertidaksamaan *constraint* (2.43) atau

$$\min_{\underline{\theta}} \frac{1}{2} \underline{\delta}^T \underline{\Phi} \underline{\delta} + \underline{\phi} \underline{\delta} \quad (2.43)$$

berdasarkan pada *constraints*

$$\underline{\Omega} \underline{\delta} \leq \underline{\omega} \quad (2.44)$$

Bentuk (2.42) dan (2.44) adalah masalah optimasi standar yang disebut sebagai permasalahan *Quadratic Programming* (QP). Bila ada bagian yang aktif di dalam himpunan *constraints* pada persamaan (2.44), maka bagian aktif tersebut akan membuat pertidaksamaan (2.44) menjadi suatu persamaan

$$\underline{\Omega}_a \underline{\delta} = \underline{\omega}_a \quad (2.45)$$

dengan matriks $\underline{\Omega}_a$ adalah bagian yang aktif dari matriks pertidaksamaan (2.44). Persamaan (2.45) kemudian dijadikan sebagai *constraints* dari fungsi kriteria pada persamaan (2.43).

Permasalahan optimasi persamaan (2.43) dengan subyek terhadap persamaan (2.45) dapat diselesaikan dengan teori pengali *Lagrange*

$$\min_{\underline{\delta}, \underline{\lambda}} L(\underline{\delta}, \underline{\lambda}) \quad (2.46)$$

dengan

$$L(\underline{\delta}, \underline{\lambda}) = \frac{1}{2} \underline{\delta}^T \underline{\Phi} \underline{\delta} + \underline{\phi} \underline{\delta} + \underline{\lambda} (\underline{\Omega}_a \underline{\delta} - \underline{\omega}_a) \quad (2.47)$$

Selanjutnya dengan melakukan diferensiasi parsial terhadap $\underline{\delta}$ dan $\underline{\lambda}$ dari persamaan (2.47), maka didapatkan kondisi *Karush-Kuhn-Tucker* sebagai berikut

$$\nabla_{\underline{\delta}} L(\underline{\delta}, \underline{\lambda}) = \underline{\Phi} \underline{\delta} + \underline{\phi} + \underline{\Omega}_a^T \underline{\lambda} \quad (2.48)$$

$$\nabla_{\underline{\lambda}} L(\underline{\delta}, \underline{\lambda}) = \underline{\Omega}_a \underline{\delta} - \underline{\omega}_a \quad (2.49)$$

atau

$$\nabla L(\underline{\delta}, \underline{\lambda}) = \begin{bmatrix} \underline{\Phi} & \underline{\Omega}_a^T \\ \underline{\Omega}_a & \underline{0} \end{bmatrix} \begin{bmatrix} \underline{\delta} \\ \underline{\lambda} \end{bmatrix} - \begin{bmatrix} -\underline{\phi} \\ \underline{\omega}_a \end{bmatrix} \quad (2.50)$$

Selanjutnya dengan membuat $\nabla L(\underline{\delta}, \underline{\lambda}) = 0$, maka didapatkan solusi optimal untuk $\underline{\delta}$ dan $\underline{\lambda}$ sebagai berikut

$$\begin{bmatrix} \underline{\delta} \\ \underline{\lambda} \end{bmatrix}_{opt} = \begin{bmatrix} \underline{\Phi} & \underline{\Omega}_a^T \\ \underline{\Omega}_a & \underline{0} \end{bmatrix}^{-1} \begin{bmatrix} -\underline{\phi} \\ \underline{\omega}_a \end{bmatrix} \quad (2.51)$$

Solusi pada *Quadratic Programming* pada kondisi normal menghasilkan nilai yang *feasible*, yaitu nilai yang memenuhi pertidaksamaan *constraints* yang ada dan dapat menghasilkan nilai fungsi kriteria minimum. Masalah yang paling sering muncul pada optimasi dengan *constraints* adalah solusi yang *infeasible*, dimana nilai yang dihasilkan tidak memenuhi pertidaksamaan *constraints* yang ada. *QP solver* akan menghentikan proses perhitungan jika terjadi solusi yang *infeasible*. Hal ini tentu tidak dapat diterima karena sinyal kendali hasil

komputasi harus selalu ada untuk digunakan sebagai masukan bagi *plant*, sehingga sangat penting untuk membuat metode cadangan dalam menghitung sinyal masukan ketika algoritma MPC diterapkan.

BAB 3

PERANCANGAN PENGENDALI MPC

UNTUK SISTEM PEMFC

3.1. Sistem Dinamik Proton Exchange Membrane Fuel Cell

Fuel cell ialah alat konversi elektrokimia yang berfungsi mengubah energi kimia dari fuel (bahan bakar) menjadi energi listrik. Fuel Cell memiliki beberapa keunggulan lain yaitu :

1. Mempunyai kemampuan untuk mengkonversi energi kimia menjadi energi listrik dengan lebih sempurna (efisiensi tinggi, panas buangan dapat digunakan kembali).
2. Ramah lingkungan. Fuel cell beroperasi dengan emisi rendah, tanpa bising dan hasil buangan yang tidak berbahaya (air dan panas).
3. Fuel cell dapat ditempatkan diberbagai lokasi sesuai kebutuhan, ukurannya fleksibel.

Namun, fuel cell juga memiliki beberapa kekurangan, diantaranya;

1. Harga pasaran yang relative masih tinggi
2. Hidrogen yang akan digunakan sebagai bahan bakar tidak tersedia dengan mudah

Tiap unit sistem fuel cell terdiri atas 4 komponen utama, yaitu:

1. **Anoda** (fuel electrode) yaitu komponen yang menjadi tempat untuk bertemunya fuel dengan elektrolit, sehingga is menjadi katalisator dalam reaksi reduksi bahan bakar dan kemudian mengalirkan elektron dari reaksi tersebut menuju rangkaian eksternal (beban)
2. **Katoda** (oxygen electrode) yaitu komponen yang menjadi tempat untuk bertemunya oksigen dengan elektrolit, sehingga is menjadi katalisator dalam reaksi oksidasi oksigen dan kemudian mengalirkan elektron dari rangkaian eksternal kembali kea dalam fuel cell yang akhirnya menghasilkan air dan panas.

3. **Elektrolit** yaitu bahan yang berfungsi sebagai penghantar yang mengalirkan ion yang berasal dari bahan bakar di anoda menuju katoda.

4. **Katalis** yaitu material atau bahan khusus untuk mempercepat reaksi kimiawi atau reduksi-oksidasi.

Tekanan parsial dari hidrogen, oksigen dan air pada sisi katoda didefinisikan sebagai variabel keadaan (state variable) sistem. Pada hukum gas ideal dinyatakan bahwa tekanan parsial setiap gas adalah proporsional terhadap banyaknya gas dalam cell, yang mana dari ketiga kontribusi tersebut relevan dan tergantung dari flow rate masukan, konsumsi gas dan flow rate keluaran gas [1].

Dari persamaan perubahan tekanan parsial gas-gas yang digunakan, dapat dituliskan persamaan variabel keadaan dan keluaran model nonlinier sistem dinamik fuel cell PEM, yang dinyatakan dalam persamaan (3.1) dan persamaan (3.2) :

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} \frac{RT}{V_A} \left(1 - \frac{x_1}{P_{op}}\right) \\ 0 \\ 0 \end{bmatrix} u_1 + \begin{bmatrix} 0 \\ \frac{RT}{V_C} \left(1 - \frac{x_2}{P_{op}}\right) \\ -\frac{RT}{V_C P_{op}} x_3 \end{bmatrix} u_2 + \begin{bmatrix} \frac{RT}{V_A} \left(-2K_r A_c \left(1 - \frac{x_1}{P_{op}}\right)\right) \\ \frac{RT}{V_C} \left(-K_r A_c \left(1 - \frac{x_2}{P_{op}}\right)\right) \\ \frac{RT}{V_C} \left(2K_r A_c \left(1 - \frac{x_3}{P_{op}}\right)\right) \end{bmatrix} u_3 \quad (3.1)$$

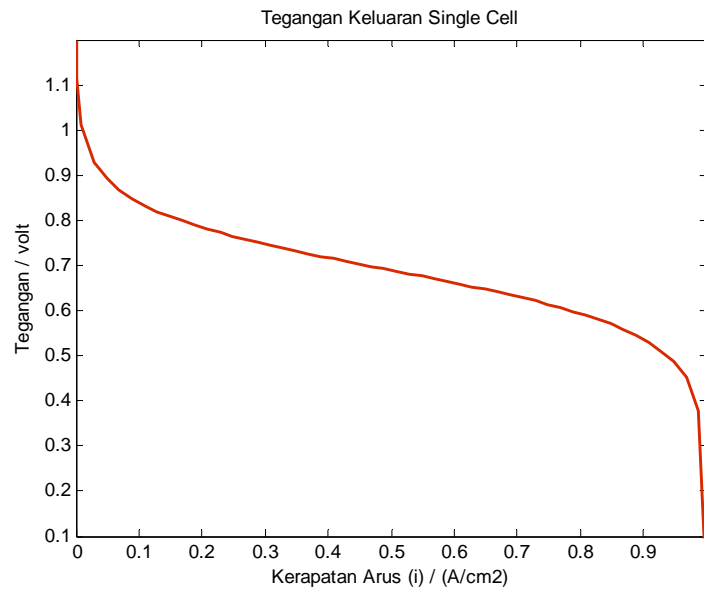
$$y = V = N \left(E^0 + \frac{RT}{2F} \ln \left\{ \frac{pH_2 (pO_2 / P_{std})^{0.5}}{pH_2O} \right\} - L \right) \quad (3.2)$$

dimana :

$$x = [pH_2 \quad pO_2 \quad pH_2O_c]^T$$

$$u = [H_{2-in} \quad O_{2-in} \quad i]^T$$

$$y = V$$



Gambar 3.1. Karakteristik tegangan Fuel-Cell pada satu sel.

3.2. Model Linier PEM FC

Untuk mendapatkan model linier yang paling tepat mewakili sistem PEMFC didapatkan pada penelitian sebelumnya. Di mana model matematik PEMFC di linierkan dan diidentifikasi untuk mencari kecocokan model yang paling sesuai. Hasil penelitian menunjukkan bahwa model linier yang dianggap dapat mewakili sistem yg sebenarnya adalah model yang turunkan melalui sistem identifikasi Least Square yaitu model state space dengan penambahan vektor kompensasi pada persamaan keadaan dan keluaran. Oleh karena itu, pada perancangan algoritma pengendali MPC, model tersebut yang akan digunakan sebagai model acuan. Berikut ini adalah persamaan state variable model;

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 0.9981 & 0.0033 & 0.0052 \\ -0.0004 & 1.0007 & 0.0014 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 41.5927 & 0.9272 & -0.3597 \\ -0.3922 & 21.8817 & -0.0507 \\ -0.0201 & 0.0086 & 0.5496 \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \end{bmatrix} + \begin{bmatrix} -0.1452 \\ -0.0506 \\ -0.0025 \end{bmatrix} \quad (3.3)$$

$$y(k) = \begin{bmatrix} 0.0765 & -0.1523 & 0.0802 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 2.6869 & -2.119 & -10.3971 \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \end{bmatrix} + 29.0544$$

Masukan sistem pada model yang sebenarnya adalah flow H2 dan flow O2, sedangkan masukan ketiga yaitu kerapatan arus (i), merupakan feedback dari tegangan keluaran sistem yang telah dilewatkan pada beban. Sehingga, pengendali MPC hanya akan mengeluarkan dua buah sinyal kendali yaitu flow H2 (u_1) dan flow O2 (u_2), dan kerapatan arus menjadi fungsi gangguan df . Dengan demikian, bentuk persamaan dari model linier berubah menjadi;

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_3(k+1) \end{bmatrix} = \begin{bmatrix} 0.9981 & 0.0033 & 0.0052 \\ -0.0004 & 1.0007 & 0.0014 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 41.5927 & 0.9272 \\ -0.3922 & 21.8817 \\ -0.0201 & 0.0086 \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + \begin{bmatrix} -0.3597 \\ -0.0507 \\ 0.5496 \end{bmatrix} df(k) + \begin{bmatrix} -0.1452 \\ -0.0506 \\ -0.0025 \end{bmatrix}$$

$$y(k) = \begin{bmatrix} 0.0765 & -0.1523 & 0.0802 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_3(k) \end{bmatrix} + \begin{bmatrix} 2.6869 & -2.119 \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix} + -10.3971.df(k) + 29.0544$$

$$x = [pH_2 \quad pO_2 \quad pH_2O_c]^T$$

$$u = [H_{2-in} \quad H_{O_2-in}]^T$$

$$df = i$$

$$y = V$$

3.2.1. Uji Observability dan Controllability Sistem

Pengujian Observability dan Controllability terhadap suatu model merupakan langkah penting yang harus dilakukan terlebih dahulu sebelum menentukan layak tidaknya sebuah model untuk dijadikan sebagai dasar perancangan sebuah pengendali. Uji *observability* sistem dimaksudkan untuk mengetahui apakah sistem tersebut benar-benar dapat diobservasi dan untuk mengetahui apakah *state-state* yang diobservasi tersebut dapat mewakili keadaan sistem yang sebenarnya. Sedangkan uji *controllability* akan menentukan apakah sistem yang diwakili oleh model tersebut dapat dikendalikan oleh sebuah pengendali.

Asumsikan bahwa model ruang keadaan sistem *PEMFC* yang ditunjukkan pada persamaan (3.3) dapat diwakili oleh persamaan berikut :

$$x((k+1)T) = Ax(kT) + Bu(kT) \quad (3.4)$$

$$y(kT) = Cx(kT) \quad (3.5)$$

Untuk melakukan pengujian *observability* dari suatu sistem, langkah yang harus dilakukan adalah membentuk matriks *observability* seperti yang ditunjukkan oleh persamaan berikut :

$$\begin{bmatrix} C^T : A^T C^T : \dots : (A^T)^{n-1} C^T \end{bmatrix} \quad (3.6)$$

dimana :

- n adalah jumlah *state* yang dimiliki oleh sebuah sistem.
- sistem *observable* jika matriks *observability* memiliki *rank* sebanyak n (jumlah *state*)

Dengan menggunakan matriks C dan A persamaan (3.5.) ke dalam persamaan (3.6), didapatkan matriks *observability* berikut ini;

$$obsv = \begin{bmatrix} 0.0765 & -0.1523 & 0.0802 \\ 0.076416 & -0.15215 & 0.080385 \\ 0.076331 & -0.15201 & 0.080569 \end{bmatrix}$$

Rank dari matriks *obsv* adalah 3, dengan Rank matriks A dari model. Hal ini menunjukkan bahwa sistem bersifat *Observable sempurna*.

Untuk melakukan pengujian *controllability* dari suatu sistem, langkah yang harus dilakukan adalah membentuk matriks *controllability* seperti yang ditunjukkan oleh persamaan berikut :

$$[B : AB : \dots : A^{n-1} B] \quad (3.7)$$

dimana :

- n adalah jumlah *state* yang dimiliki oleh sebuah sistem.
- sistem *controllable* jika matriks *controllability* memiliki *rank* sebanyak n (jumlah *state*)

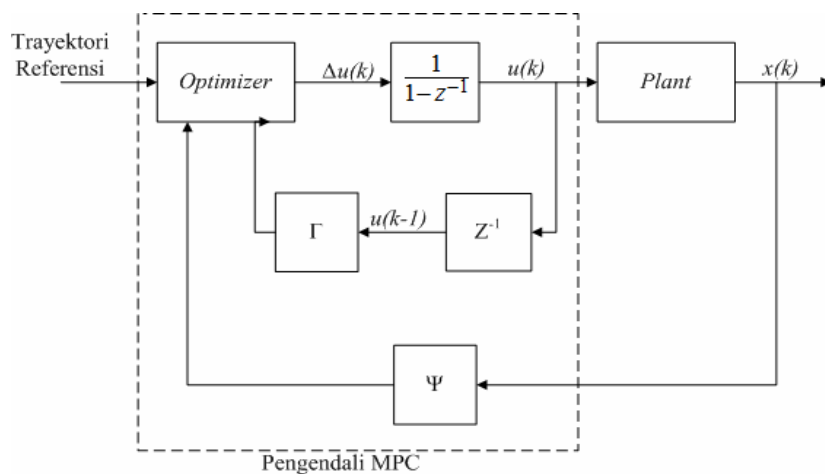
Dengan menggunakan matriks B dan A persamaan (3.5.) ke dalam persamaan (3.7), didapatkan matriks *controllability* berikut ini;

$$ctrb = \begin{bmatrix} 41.593 & 0.9272 & -0.3597 & 41.512 & 0.99769 & -0.35633 & 41.432 & 1.0681 & -0.35296 \\ -0.3922 & 21.882 & -0.0507 & -0.40914 & 21.897 & -0.049822 & -0.42606 & 21.912 & -0.048945 \\ -0.0201 & 0.0086 & 0.5496 & -0.0201 & 0.0086 & 0.5496 & -0.0201 & 0.0086 & 0.5496 \end{bmatrix}$$

Rank dari matriks $ctrb$ adalah 3, dengan Rank matriks A dari model. Hal ini menunjukkan bahwa sistem bersifat *controllable sempurna*.

3.3. Algoritma Model Predictive Control dengan Constraints

Struktur pengendali MPC dengan *constraint* untuk model ruang keadaan terdapat pada gambar 3.14. Dari blok diagram tersebut, terlihat bahwa prediksi perubahan sinyal masukan sekarang ($\Delta u(k)$) membutuhkan data dari variabel keadaan sekarang $\underline{x}(k)$ dan masukan satu langkah sebelumnya $u(k-1)$.



Gambar 3.2. Blok diagram pengendali MPC dengan *constraints*.

Algoritma perhitungan perubahan sinyal kendali pada MPC dengan *constraints* adalah sebagai berikut :

1. Parameter pengendali yang terlebih dahulu harus ditentukan antara lain horizon prediksi (H_p), horizon kendali (H_u), matriks faktor bobot kesalahan (Q), dan matriks faktor bobot perubahan sinyal kendali (R).
2. Matriks $\underline{\mathcal{F}}$ dihitung dengan menggunakan persamaan (2.25), serta matriks $\underline{\mathcal{H}}$ dan $\underline{\mathcal{G}}$ yang terdapat pada fungsi kriteria persamaan (2.29) dihitung masing-masing dengan menggunakan persamaan (2.31) dan (2.30).

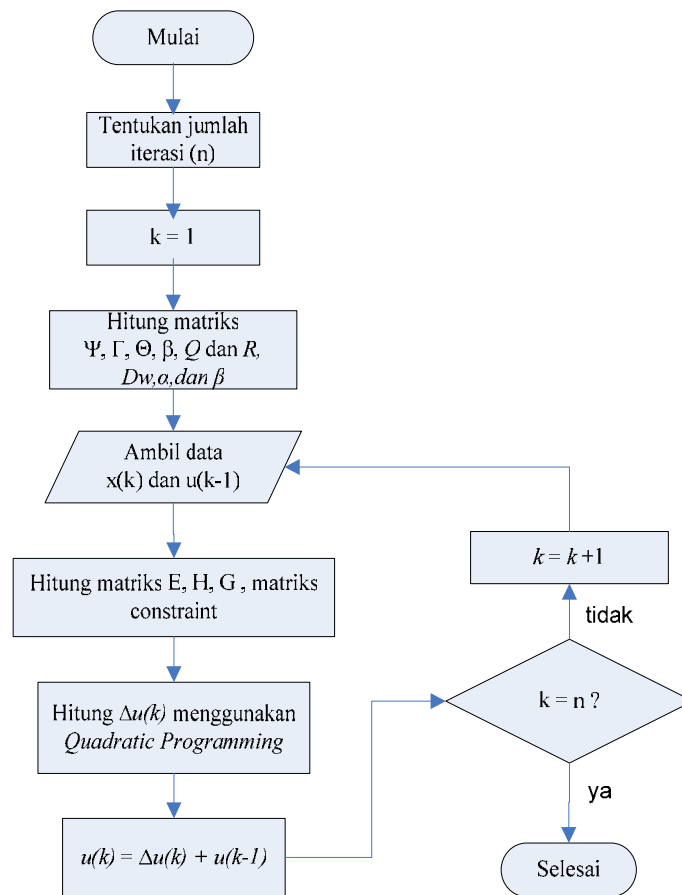
- Parameter batasan (*constraints*) fisik sistem diubah ke dalam bentuk pertidaksamaan yang memiliki hubungan dengan perubahan sinyal kendali (ΔU).

$$\underline{\Omega} \Delta U(k) \leq \underline{\omega} \quad (3.8)$$

- Menghitung perubahan sinyal kendali optimal Δu_{opt} dengan menggunakan metode *Quadratic Programming*.
- Menghitung sinyal kendali $u(k)$ dimana

$$u(k) = \Delta u(k) + u(k-1) \quad (3.9)$$

Diagram alir untuk perhitungan sinyal kendali dengan menggunakan MPC dengan *constraints* adalah seperti pada gambar 3.3.



Gambar3.3. Diagram alir algoritma MPC dengan *constraints*.

Metode yang digunakan pada *Quadratic Programming* dalam menghitung nilai $\underline{\Delta U}$ adalah *Active Set* dengan alur operasi seperti dijelaskan berikut ini [2].

1. Fungsi kriteria pada persamaan (2.37), diubah menjadi seperti berikut

$$V(\underline{\Delta U}(k)) = \frac{1}{2} \underline{\Delta U}^T(k) 2\underline{\mathcal{H}} \underline{\Delta U}(k) - \underline{\Delta U}^T(k) \underline{\mathcal{G}} \quad (3.10)$$

berdasarkan *constraints*

$$\underline{\Omega} \underline{\Delta U}(k) \leq \underline{\omega} \quad (3.11)$$

2. Nilai $\underline{\Delta U}_r$ dipilih sedemikian sehingga pertidaksamaan *constraints* (3.11) menjadi sebuah persamaan seperti berikut

$$\underline{\Omega}_r \underline{\Delta U}_r = \underline{\omega}_r \quad (3.12)$$

Elemen yang membuat pertidaksamaan menjadi persamaan disebut elemen aktif .

3. Menghitung nilai \underline{d} yang merupakan pergerakan $\underline{\Delta U}_r$ dalam meminimasi fungsi kriteria sehingga fungsi kriteria pada persamaan (3.10) berubah menjadi

$$\begin{aligned} V(\underline{\Delta U}_r + \underline{d}) &= \frac{1}{2} (\underline{\Delta U}_r + \underline{d})^T 2\underline{\mathcal{H}} (\underline{\Delta U}_r + \underline{d}) - (\underline{\Delta U}_r + \underline{d})^T \underline{\mathcal{G}} \\ &= \frac{1}{2} \underline{d}^T \underbrace{2\underline{\mathcal{H}} \underline{d}}_{\underline{\Phi}} + \underline{d}^T \underbrace{(2\underline{\mathcal{H}} \underline{\Delta U}_r - \underline{\mathcal{G}})}_{\underline{\phi}_r} + V(\underline{\Delta U}_r) \end{aligned} \quad (3.13)$$

Nilai \underline{d} tidak boleh mempengaruhi pertidaksamaan *constraints* (3.11), sehingga persamaan *constraints* untuk persamaan (3.13) adalah

$$\underline{\Omega}_r \underline{d} = \underline{0} \quad (3.14)$$

4. Dari persamaan (3.13) dan (3.14), nilai optimal \underline{d} sepanjang *constraints* yang aktif dapat dihitung dengan menyelesaikan fungsi kuadrat berikut

$$\min \frac{1}{2} \underline{d}^T \underline{\Phi} \underline{d} + \underline{d}^T \underline{\phi}_r \quad (3.15)$$

dengan *constraints*

$$\underline{\Omega}_r \underline{d} = \underline{0} \quad (3.16)$$

Nilai pengali *Lagrange* $\underline{\lambda}_r$ untuk persamaan (3.15) dan (3.16) dihitung berdasarkan kondisi *Karush-Kuhn-Tucker* (KKT) seperti berikut

$$\begin{bmatrix} \underline{\Phi} & \underline{\Omega}^T \\ \underline{\Omega} & \underline{0} \end{bmatrix} \begin{bmatrix} \underline{d} \\ \underline{\lambda}_r \end{bmatrix} = \begin{bmatrix} -\underline{\phi}_r \\ \underline{0} \end{bmatrix} \quad (3.17)$$

dimana nilai $\underline{\Delta U}_r$ yang terdapat pada matriks $\underline{\phi}_r$ ditentukan pada langkah (2). Hasil perhitungan \underline{d} dan $\underline{\lambda}_r$ akan mempengaruhi tahapan berikutnya, yaitu :

- a. Jika semua $\underline{\lambda}_r > 0$ dan $\underline{d} = 0$, maka proses komputasi selesai dan nilai $\underline{\Delta U}_r$ merupakan nilai optimal untuk $\underline{\Delta U}(k)$.
- b. Jika semua $\underline{\lambda}_r > 0$ dan ada nilai $d \neq 0$, maka lanjut ke langkah (5).
- c. Jika ada nilai $\lambda_r < 0$, maka *constraint* yang memiliki nilai λ_r paling negatif dibuang, kemudian lanjut ke langkah (5).

5. Nilai faktor koreksi pergerakan nilai optimal α_r dihitung dengan menggunakan rumus

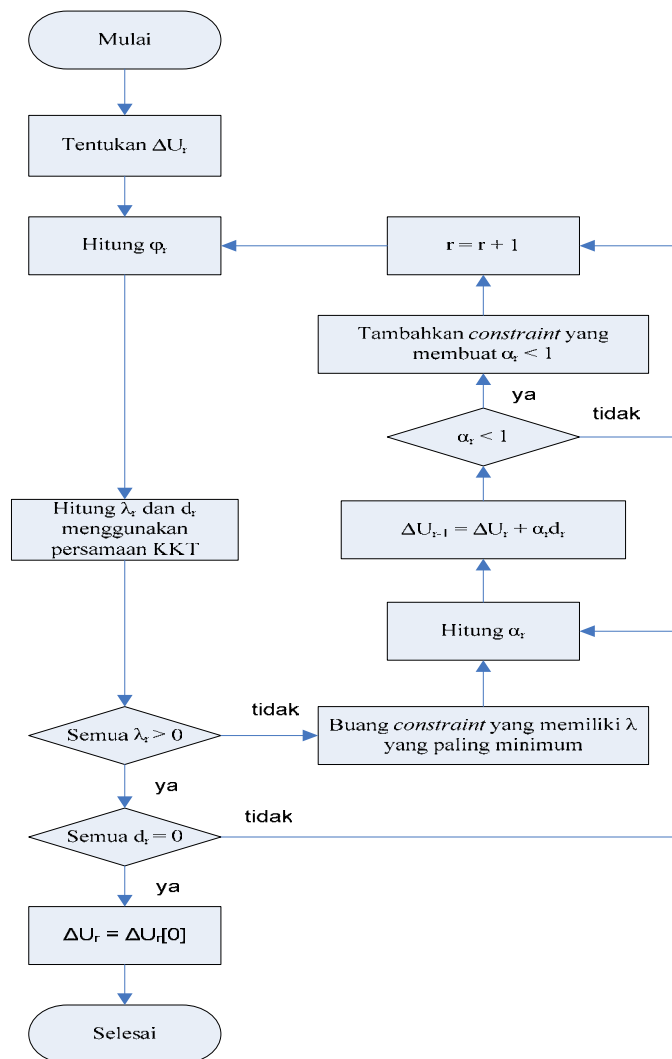
$$\alpha_r = \min \left(1, \min_{\substack{i \in \underline{\Omega}_r \\ a_i d > 0}} \frac{b_i - a_i \underline{\Delta U}_r}{a_i d} \right) \quad (3.18)$$

dengan a_i adalah baris dari pertidaksamaan batasan yang tidak aktif dan b_i adalah batasannya. Selanjutnya, nilai $\underline{\Delta U}_r$ dalam arah \underline{d} dihitung sebagai berikut

$$\underline{\Delta U}_{r+1} = \underline{\Delta U}_r + \alpha_r \underline{d} \quad (3.19)$$

6. Jika nilai $\alpha_r < 1$, maka *constraint* yang membuat nilai $\alpha_r < 1$ ditambahkan ke \underline{Q}_r .
7. Tetapkan $r = r + 1$ dan kembali ke langkah (3) untuk proses iterasi berikutnya.

Diagram alir metode *Active Set* untuk menyelesaikan *Quadratic Programming* seperti yang terdapat pada gambar 3.4.



Gambar 3.4. Diagram alir metode *Active Set* untuk menyelesaikan *Quadratic Programming*.

3.4. Penurunan Persamaan Sinyal Kendali

Struktur model yang akan dikendalikan:

$$\begin{aligned}\underline{x}(k+1) &= \underline{A}\underline{x}(k) + \underline{B}\underline{u}(k) + B_d df(k) + K_x \\ \underline{y}(k) &= C\underline{x}(k) + \underline{D}\underline{u}(k) + D_d df(k) + K_y\end{aligned}\quad (3.20)$$

Berikut ini adalah gambaran umum perhitungan variabel keadaan dan keluaran prediksi dengan menggunakan MPC pada model yang mengandung gangguan pada persamaan variabel keadaan dan keluarannya. Karena model linier yang digunakan mengandung matriks D (directfeedthrough) pada persamaan keluarannya, maka panjang horizon untuk prediksi dan sinyal kendali adalah sama ($H_p=H_u$). Untuk prediction horizon (H_p) dan control horizon (H_u) = 3, penurunan matematis untuk perhitungan error $E(k)$ ditunjukkan pada persamaan berikut ini:

$$\begin{aligned}x(k+1|k) &= Ax(k) + Bu(k) + B_d d(k) + K_x \\ x(k+2|k) &= Ax(k+1) + Bu(k+1) + B_d d(k+1) + K_x \\ &= A^2 x(k) + ABu(k) + AB_d d(k) + AK_x + Bu(k+1) + B_d d(k+1) + K_x \\ x(k+3|k) &= Ax(k+2) + Bu(k+2) + B_d d(k+2) + K_x \\ &= A^3 x(k) + A^2 Bu(k) + A^2 B_d d(k) + A^2 x(k) + ABu(k+1) + AB_d d(k+1) + AK_x + Bu(k+2) + B_d d(k+2) + K_x \\ \hat{y}(k+1|k) &= Cx(k+1) + Du(k+1) + D_d d(k+1) + K_y \\ &= CAx(k) + CBu(k) + CB_d d(k) + CK_x + Du(k+1) + D_d d(k+1) + K_y \\ \hat{y}(k+2|k) &= Cx(k+2) + Du(k+2) + D_d d(k+2) + K_y \\ &= CA^2 x(k) + CABu(k) + CAB_d d(k) + CAK_x + CBu(k+1) + CB_d d(k+1) + CK_x + Du(k+2) + D_d d(k+2) + K_y \\ \hat{y}(k+3|k) &= Cx(k+3) + Du(k+3) + D_d d(k+3) + K_y \\ &= CA^3 x(k) + CA^2 Bu(k) + CA^2 B_d d(k) + CA^2 x(k) + CABu(k+1) + CAB_d d(k+1) + CAK_x + CBu(k+2) + CB_d d(k+2) \\ &\quad + CK_x + Du(k+3) + D_d d(k+3) + K\end{aligned}$$

dimana

$$\begin{aligned}u(k|k) &= \Delta u(k|k) + u(k-1) \\ u(k+1|k) &= \Delta u(k+1|k) + \Delta u(k|k) + u(k-1) \\ u(k+2|k) &= \Delta u(k+2|k) + \Delta u(k+1|k) + \Delta u(k|k) + u(k-1) \\ u(k+i|k) &= u(k+H-1|k), \text{ for } i \geq H\end{aligned}$$

$$\begin{aligned}d(k|k) &= y(k) - \hat{y}(k|k-1) \\ d(k|k) &= d(k+1|k) = d(k+2|k);\end{aligned}$$

(3.21)

Dari penjabaran persamaan diatas, dapat dibuat hubungan dalam bentuk perkalian dan penjumlahan matriks untuk persamaan keluaran sebagai berikut;

$$\begin{aligned}
 \begin{bmatrix} y(k+1|k) \\ y(k+2|k) \\ y(k+3|k) \end{bmatrix} &= \begin{bmatrix} CA \\ CA^2 \\ CA^3 \end{bmatrix} x(k) + \begin{bmatrix} CB+D \\ CAB+CB+D \\ CA^2B+CB+CB+D \end{bmatrix} u(k-1) + \begin{bmatrix} CB+D & D & 0 \\ CAB+CB+D & CB+D & D \\ CA^2B+CB+CB+D & CAB+CB+D & D \end{bmatrix} \begin{bmatrix} \Delta u(k|k) \\ \Delta u(k+1|k) \\ \Delta u(k+2|k) \end{bmatrix} \\
 &\quad C_y \varphi \qquad \qquad \qquad C_y \gamma \qquad \qquad \qquad C_y \theta \\
 &+ \begin{bmatrix} CB_d & D_d & 0 \\ CAB_d & CB_d & D_d \\ CA^2B_d & CAB_d & CB_d + D_d \end{bmatrix} \begin{bmatrix} d(k|k) \\ d(k+1|k) \\ d(k+2|k) \end{bmatrix} + \begin{bmatrix} CK_x \\ CAK_x + CK_x \\ CA^2K_x + CAK_x + CK_x \end{bmatrix} + \begin{bmatrix} K_y \\ K_y \\ K_y \end{bmatrix} \\
 &\quad C_y D_w \qquad \qquad \qquad C_y \alpha \qquad \qquad \qquad \beta
 \end{aligned} \tag{3.22}$$

Persamaan matematis dari matriks $\underline{\mathcal{E}}(k)$ adalah sebagai berikut

$$\underline{\mathcal{E}}(k) = \underline{T}(k) - \underline{C}_y \underline{\Psi} x(k) - \underline{C}_y \underline{\Gamma} u(k-1) - \underline{C}_y \underline{D}_w d(k) - \underline{C}_y \underline{\alpha} - \beta \tag{3.23}$$

Perhitungan selanjutnya sesuai dengan urutan strategi perhitungan sinyal kendali yang telah diuraikan pada subbab 2.1.

3.5. Langkah Perhitungan Sinyal Kendali untuk Sistem PEMFC

Berikut ini adalah contoh langkah-langkah yang dilakukan untuk perhitungan sinyal kendali dengan metode MPC dengan constraints. Parameter-parameter pengendali yang digunakan adalah sebagai berikut;

- Nilai *control horizon* (H_u) = 3
- Nilai *prediction horizon* (H_p) = 3
- Faktor Bobot Kesalahan $\underline{Q} = \underline{I}_{H_p}$
- Faktor Bobot perubahan sinyal kendali $\underline{R} = \underline{I}_{H_u}$
- Trayektori acuan (set point) $r(k) = 21$
- Batasan (constraint) yang digunakan:

$$-0.1 \leq \Delta u(k) \leq 0.1$$

$$0.001 \leq u(k) \leq 0.55$$

o Matriks variable keadaan:

$$A = \begin{bmatrix} 0.9981 & 0.0033 & 0.0052 \\ -0.0004 & 1.0007 & 0.0014 \\ 0 & 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 41.5927 & 0.9272 \\ -0.3922 & 21.8817 \\ -0.0201 & 0.0086 \end{bmatrix}, Bd = \begin{bmatrix} -0.3597 \\ -0.0507 \\ 0.5496 \end{bmatrix}$$

$$C = [0.0765 \quad -0.1523 \quad 0.0802], D = [2.6869 \quad -2.119], Dd = [-10.3971]$$

$$Kx = \begin{bmatrix} -0.1452 \\ -0.0506 \\ -0.0025 \end{bmatrix}, Ky = [29.0544]$$

Untuk mendapatkan sinyal kendali, algoritma MPC dengan constraint melakukan tahapan perhitungan sebagai berikut;

1. Menghitung matriks yang ada dalam persamaan ruang keadaan MPC, yaitu matriks $\underline{C}_y \underline{\Psi}$, $\underline{C}_y \underline{\Gamma}$, $\underline{C}_y \underline{\Theta}$, $\underline{C}_y \underline{D}_w$, $\underline{C}_y \underline{\alpha}$ dan β dengan menggunakan persamaan (3.22)

$$\underline{C}_y \underline{\Psi} = \begin{bmatrix} 0.076416 & -0.15215 & 0.080385 \\ 0.076331 & -0.15201 & 0.080569 \\ 0.76247 & -0.15186 & 0.080753 \end{bmatrix} \quad \underline{C}_y \underline{\Gamma} = \begin{bmatrix} 5.9269 & -5.38 \\ 9.1633 & -8.6378 \\ 12.396 & -11.893 \end{bmatrix}$$

$$\underline{C}_y \underline{\Theta} = \begin{bmatrix} 5.9269 & -5.38 & 2.6869 & -2.119 & 0 & 0 \\ 9.1633 & -8.6378 & 5.9269 & -5.38 & 2.6869 & -2.119 \\ 12.396 & -11.893 & 9.1633 & -8.6378 & 5.9269 & -5.38 \end{bmatrix}$$

$$\underline{C}_y \underline{D}_w = \begin{bmatrix} 0.024282 & -10.397 & 0 \\ 0.024407 & 0.024282 & -10.397 \\ 0.024531 & 0.024407 & -10.373 \end{bmatrix}$$

$$\underline{C}_y \underline{\alpha} = \begin{bmatrix} -0.036019 \\ -0.0071994 \\ -0.010793 \end{bmatrix} \quad \underline{\alpha} = \begin{bmatrix} 29.054 \\ 29.054 \\ 29.054 \end{bmatrix}$$

2. Batasan yang telah ditentukan, terlebih dahulu dinyatakan dalam bentuk $\Delta u(k)$ berdasarkan persamaan dasar berikut;

$$u(k) = \Delta u(k) + u(k-1)$$

$$u(k+1) = \Delta u(k+1) + \Delta u(k) + u(k-1)$$

Untuk k selanjutnya, hanya dilakukan penambahan $\Delta u(k+i)$ hingga $i=Hu$

3. Matriks \underline{G} dan \underline{H} masing-masing dihitung dengan menggunakan persamaan (2.28), dan (2.29). Dengan membuat nilai matriks \underline{Q} sama dengan \underline{I}_6 dan matriks \underline{R} bernilai \underline{I}_3 , maka matriks \underline{H} dapat dihitung sebagai berikut

$$\underline{H} = (\underline{C}_y \underline{\Theta})^T \underline{Q} \underline{C}_y \underline{\Theta} + \underline{R}$$

$$\underline{H} = \begin{bmatrix} 273.76 & -258.46 & 183.82 & -168.93 & 98.091 & -86.107 \\ -258.46 & 245.99 & -174.62 & 160.6 & -93.694 & 82.285 \\ 183.82 & -174.62 & 127.31 & -116.73 & 70.234 & -61.857 \\ -168.93 & 160.6 & -116.73 & 109.05 & -65.651 & 57.871 \\ 98.091 & -93.694 & 70.234 & -65.651 & 43.347 & -37.58 \\ -86.107 & 82.285 & -61.857 & 57.871 & -37.58 & 34.434 \end{bmatrix}$$

Untuk matriks \underline{G} , akan diperbarui setiap waktu seiring dengan perubahan vektor kesalahan $\underline{E}(k)$ yang terjadi. Untuk $k=1$;

$$\underline{G} = 2(\underline{\Theta} \underline{C}_y)^T \underline{Q} \underline{E}(1)$$

$$\underline{G} = \begin{bmatrix} -39.52 \\ 37.266 \\ -25.623 \\ 23.277 \\ -12.492 \\ 10.882 \end{bmatrix}$$

Perhitungan untuk matriks \underline{Z} berdasarkan persamaan (3.23), dengan memisalkan pada $k=1$ $\underline{x}(1) = [0 \ 0]^T$, $u(0) = 0$ dan

$$\underline{T} = \begin{bmatrix} 21 \\ 21 \\ 21 \end{bmatrix}$$

maka didapatkan

$$\underline{E}(1) = \begin{bmatrix} -0.70196 \\ -0.71565 \\ -0.72944 \end{bmatrix}$$

Jika pengendali MPC yang digunakan tidak menentukan batasan apapun (MPC Noconstrained), maka setelah langkah ini, langsung digunakan persamaan (2.31) untuk perhitungan $\Delta u(k)$ optimal di setiap langkah pencuplikan. Yang diperbarui setiap waktu adalah vektor kesalahan \underline{E} , perubahan sinyal kendali $\Delta u(k)$.

4. Untuk membuat *constraints* pada algoritma MPC, di buat sebuah matriks augmented yang mengandung semua unsur sinyal kendali untuk yaitu $\Delta u(k), u(k-1), u(k)$ untuk k di sepanjang rentang control horizon yang telah ditentukan untuk semua masukan. Untuk sistem PEMFC, terdapat dua buah masukan, sehingga untuk H_u sebesar 3, matriks augmented yang dibuat adalah sebagai berikut:

$$\underline{u}_{aug} = \begin{bmatrix} \Delta u_1(k) \\ \Delta u_2(k) \\ \Delta u_1(k+1) \\ \Delta u_2(k+1) \\ \Delta u_1(k+2) \\ \Delta u_2(k+2) \\ u_1(k) \\ u_2(k) \\ u_1(k+1) \\ u_2(k+1) \\ u_1(k+2) \\ u_2(k+2) \\ u_1(k-1) \\ u_2(k-1) \end{bmatrix}$$

Berdasarkan matriks augmented ini, dibuat matriks pertidaksamaan dan persamaan yang akan berfungsi sebagai batasan perhitungan sinyal kendali sehingga berlaku hubungan berikut;

$$\underline{A}u_{avg} \leq \underline{B}$$

$$\underline{A}_{eq}u_{avg} = \underline{B}_{eq}$$

Dimana \underline{A} dan \underline{B} merupakan matriks yang berisi nilai batasan maksimum dan minimum untuk semua nilai Δu dan u yang dihitung. Untuk perancangan ini, \underline{B}_{eq} dibuat sama dengan 0, sehingga matriks \underline{A}_{eq} mengandung hubungan antara setiap komponen u_{avg} yang menghasilkan nilai 0. Dengan demikian, pengendali MPC akan mengoptimasi sinyal kendali yang dihasilkan agar memenuhi constraint persamaan dan pertidaksamaan tersebut.

$$\underline{A} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$\underline{B} = \begin{bmatrix} -(-0.1) \\ -(-0.1) \\ -(-0.1) \\ -(-0.1) \\ -(-0.1) \\ -(-0.1) \\ -0.001 \\ -0.001 \\ -0.001 \\ -0.001 \\ -0.001 \\ -0.001 \\ 0.1 \\ 0.1 \\ 0.1 \\ 0.1 \\ 0.1 \\ 0.1 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \\ 0.55 \end{bmatrix}$$

$$\underline{A}_{eq} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & -1 & 0 & 0 \end{bmatrix} \quad \underline{B}_{eq} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Jika terdapat constraint pada persamaan maka matriks ini yang berubah, misalnya untuk constraint rasio *flow* antara hydrogen dan oksigen

$$u_2(k) = 0.89 u_1(k)$$

$$\underline{A}_{eq} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1.12 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & -0.89 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1.12 & 0 & -1.12 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & -0.89 & 0 & -0.89 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1.12 & 0 & -1.12 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0.89 & 0 & -0.89 & 0 & 0 & 0 & 0 \end{bmatrix}$$

5. Untuk menghitung $\Delta u(k)$, digunakan metode *quadratic programming* untuk perhitungan sinyal kendali. Pada perancangan ini, digunakan fungsi *fmincon* yang ada dalam Matlab. Fungsi ini meminimumkan sebuah fungsi terhadap constraint yang telah ditentukan. Dalam MPC, yang diminimumkan adalah fungsi kriteria. Pada setiap pencuplikan k , akan terhitung semua unsur sinyal kendali yang terdapat pada matriks augmented, namun pada MPC, nilai yang digunakan hanya nilai perubahan sinyal kendali pada saat k ($\Delta u(k)$), sedangkan nilai yang lain tidak digunakan. Untuk memperbarui sinyal kendali, nilai $\Delta u(k)$ inilah yang akan dijumlahkan dengan nilai sinyal kendali sebelumnya ($u(k-1)$). Misalkan, untuk $k=k+1$, perhitungannya sebagai berikut;

$$u(k+1) = \Delta u(k) + u(k-1)$$

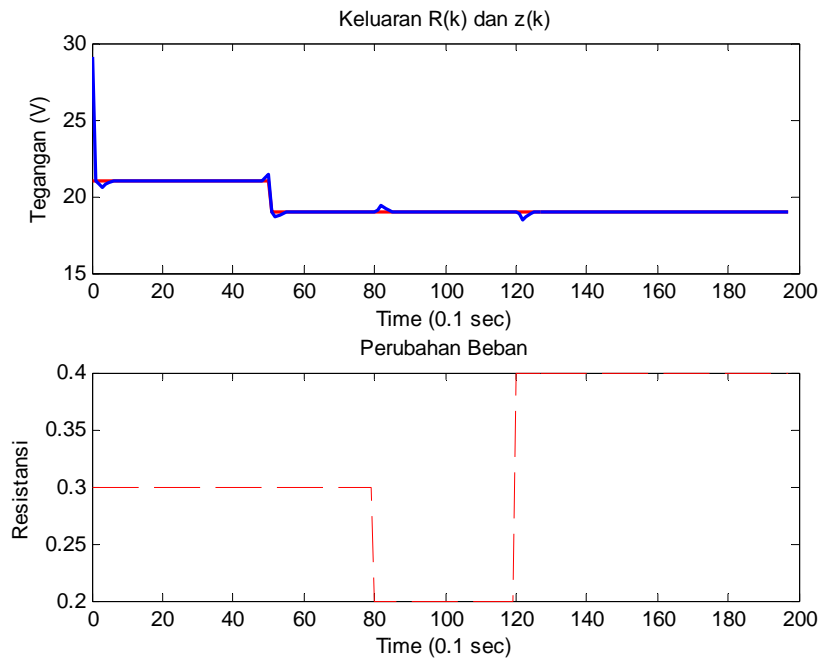
BAB 4

ANALISA HASIL PERANCANGAN

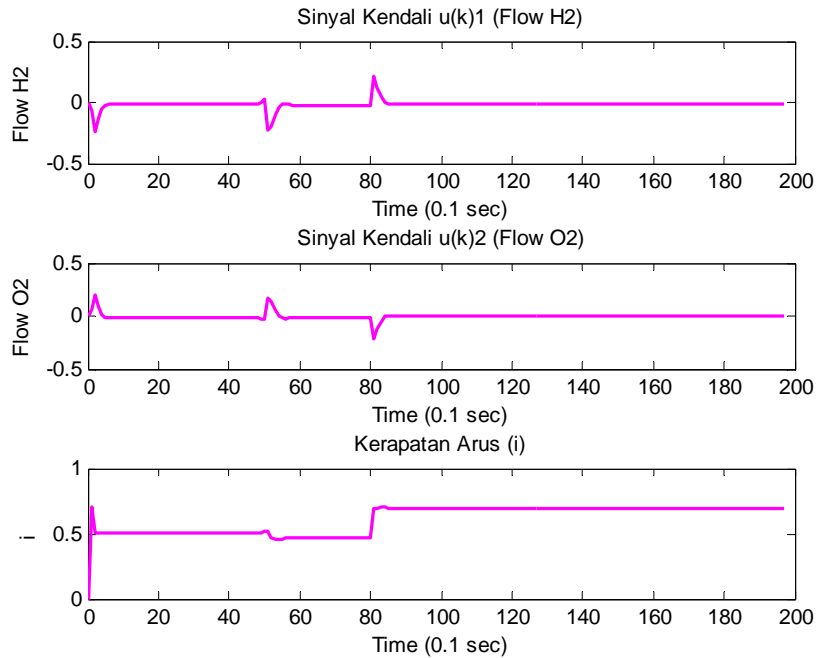
Tujuan dari uji coba dan analisa adalah untuk mengetahui kinerja dari pengendali *MPC* tanpa constraint dan dengan constraint dengan parameter penalaan yang berbeda-beda. Perancangan algoritma pengendali menggunakan program Matlab. Untuk semua tahapan pengujian yang dilakukan, model dari sistem PEMFC diberikan sinyal acuan (*trajectory*) serta sinyal gangguan (*disturbance*) berupa perubahan beban yang berubah pada waktu pencuplikan tertentu. Dengan uji ini, keterbatasan dan keandalan dari pengendali yang dirancang dapat diketahui.

4.1 Pengujian MPC Tanpa Constraint

Parameter pengendali yang digunakan adalah $H_p = H_u = 3$ dan $Q = I_3$, $R = I_6$ Gambar 4.1 dan 4.2. menunjukkan keluaran prediksi, gangguan, serta sinyal kendali yang dihasilkan.



Gambar 4.1. Keluaran sistem dengan pengendali MPC tanpa constraint



Gambar 4.2. Sinyal kendali pengendali MPC tanpa constraint

Dengan menggunakan MPC noconstraint (tanpa batasan), sinyal keluaran model dapat mengikuti perubahan sinyal acuan dan dapat mengatasi gangguan dengan baik. Namun, sinyal kendali yang dihasilkan tidak sesuai dengan karakteristik fisik dari sistem PEMFC yang sebenarnya, karena sinyal kendali bernilai ada yang bernilai negatif. Untuk pengujian selanjutnya, rasio antara besar aliran Hydrogen dan Oxygen akan menjadi constraint pada sinyal kendali.

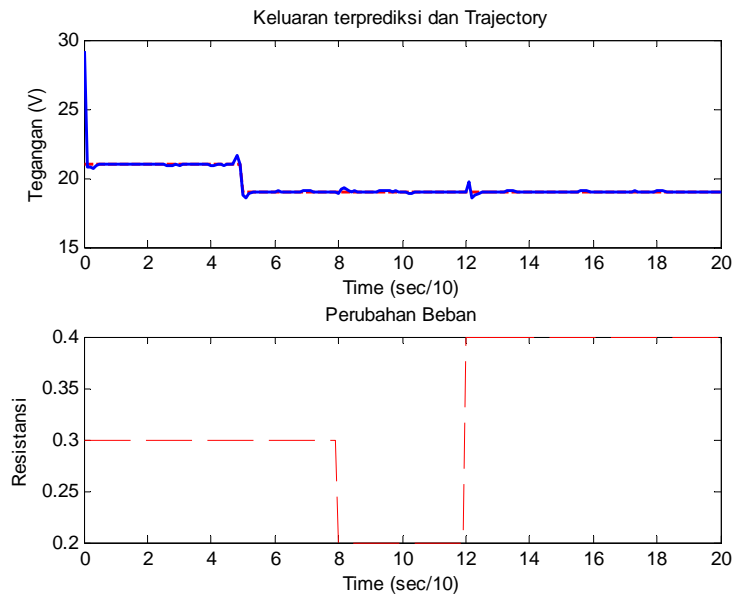
4.2 Pengujian MPC dengan constraint

Berdasarkan karakteristik fisik dari sistem PEMFC, sinyal masukan yang berupa *flow* Hydrogen dan *flow* Oxygen harus bernilai positif dan dalam batas nilai tertentu. Demikian juga dengan perubahan *flow* pada tiap waktu juga harus dibatasi besaran maksimum dan minimumnya. Constraint yang digunakan untuk kebutuhan tersebut adalah

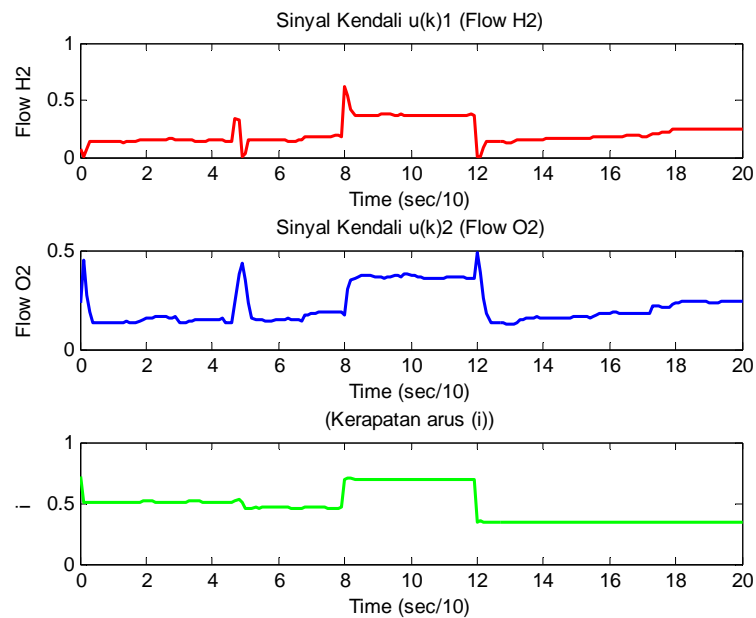
$$\begin{aligned}
 -0.1 &\leq \Delta u(k) \leq 0.1 \\
 0.001 &\leq u(k) \leq 0.55 \\
 u_2(k) &= 0.89u_1(k)
 \end{aligned}$$

4.2.1 Perbandingan hasil pengendalian MPC Constrained dengan Matriks Bobot R yang berubah-ubah

a. $Q = I_3, R = 0.1 I_6$



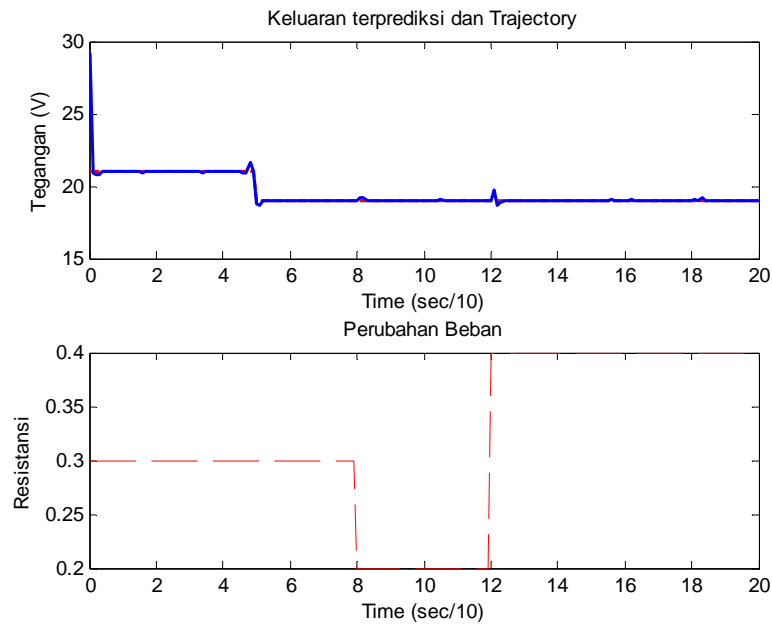
Gambar 4.3. Keluaran sistem dengan pengendali MPC constrained ($Q = I_3, R = 0.1 I_6$)



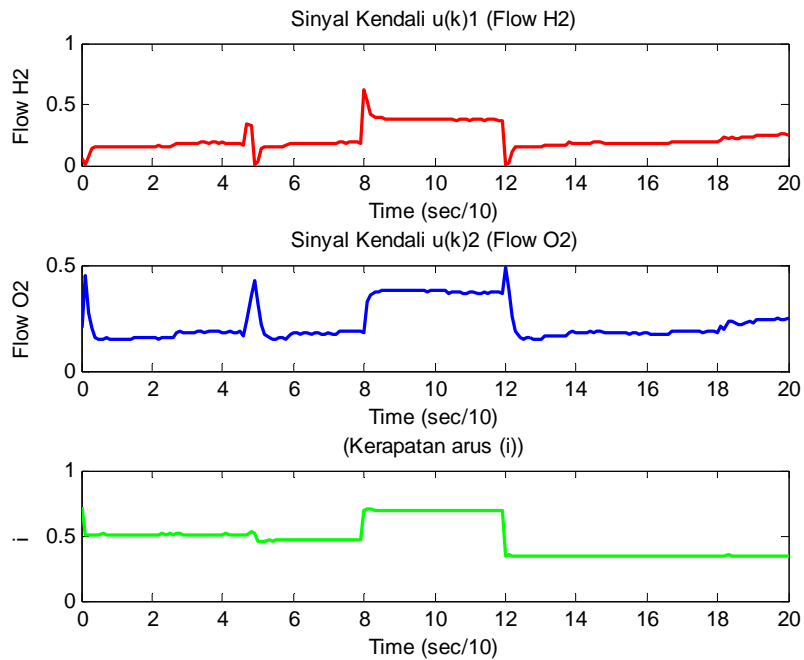
Gambar 4.4. Sinyal kendali pengendali MPC constrained ($Q = I_3, R = 0.1 I_6$)

Dengan penambahan constraint pada sinyal kendali, pengendali dapat menghasilkan sinyal kendali yang sesuai dengan karakteristik yang diinginkan yaitu berada dalam batas nilai maksimum dan minimum yang ditentukan. Meskipun masih terdapat solusi yang *infeasible* yaitu pada waktu ke 12 detik. Pada percobaan ini, nilai parameter yang digunakan $Q = I_3$ dan $R = 0.1 I_6$ menghasilkan keluaran terprediksi yang kurang halus, terdapat banyak titik dimana keluaran tidak mengikuti sinyal acuan, meskipun tanpa adanya gangguan.

b. $Q = I_3, R = 1 I_6$



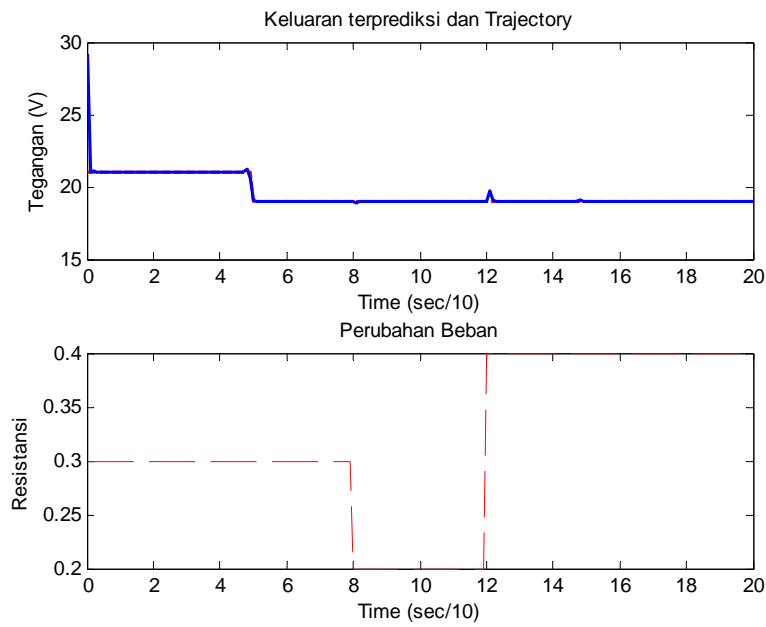
Gambar 4.5. Keluaran sistem dengan pengendali MPC constrained ($Q = I_3, R = 1 I_6$)



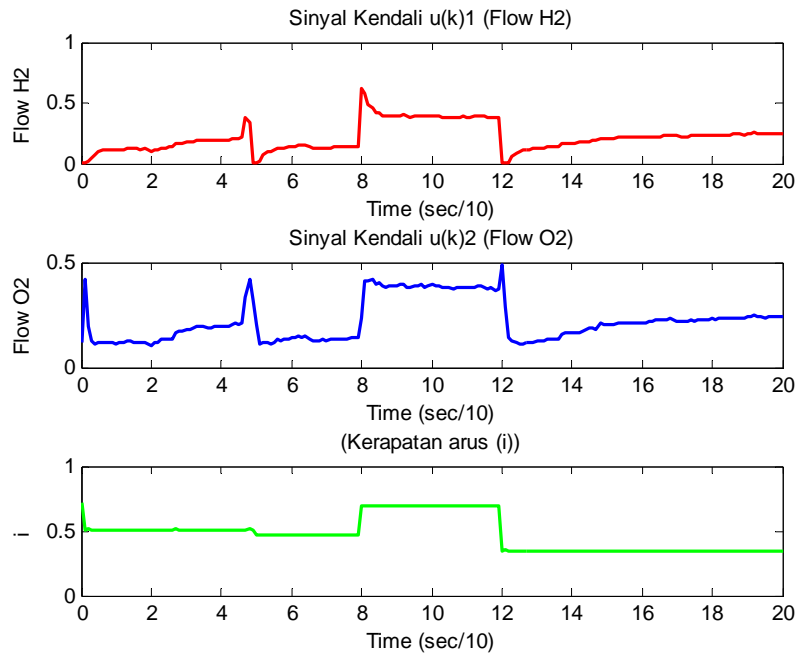
Gambar 4.6. Sinyal kendali pengendali MPC constrained
 $(Q = I_3, R = 1 I_6)$

Jika nilai parameter yang digunakan $Q = I_3$ dan $R = 1 I_6$, menghasilkan keluaran terprediksi yang lebih halus, meskipun masih terdapat beberapa titik dimana keluaran tidak mengikuti sinyal acuan meskipun tanpa adanya gangguan. Namun sinyal kendali semakin halus dengan penambahan bobot matriks R, meskipun masih terdapat solusi *infeasible* pada titik 5 detik dan 12 detik.

c. $Q = I_3, R = 10 I_6$



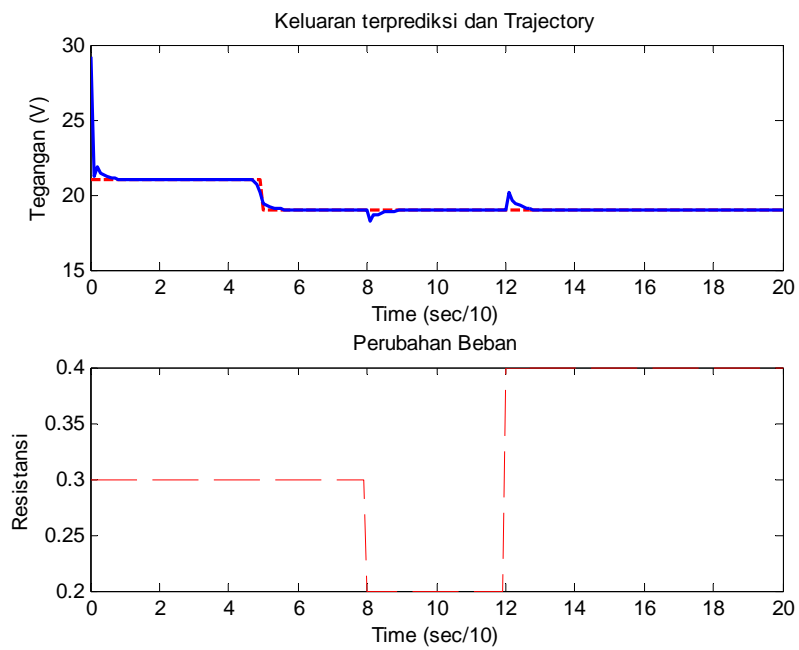
Gambar 4.7. Keluaran sistem dengan pengendali MPC constrained ($Q = I_3, R = 10 I_6$)



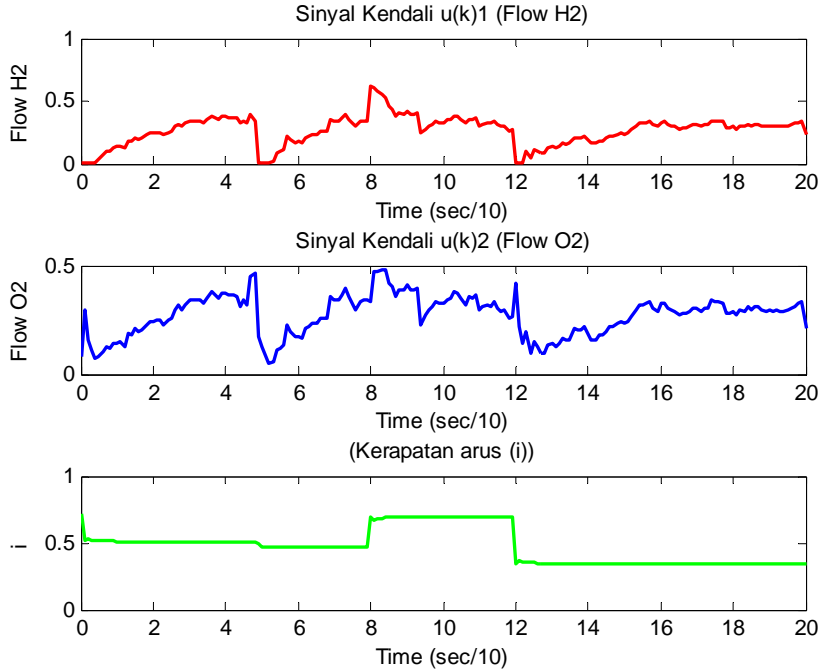
Gambar 4.8. Sinyal kendali pengendali MPC constrained ($Q = I_3, R = 10 I_6$)

Untuk $Q = I_3$ dan $R = 10 I_6$, perubahan sinyal kendali lebih di tekan sehingga dihasilkan sinyal kendali yang lebih halus dengan satu solusi *infeasible* yang sangat nampak yaitu pada titik 12 detik saat terjadi gangguan berupa perubahan beban yang besar. Keluaran prediksi juga mampu mengikuti perubahan sinyal acuan dengan baik dan hanya sedikit perubahan sinyal keluaran yang terjadi saat diberi gangguan berupa perubahan beban.

d. $Q = I_3, R = 100 I_6$



Gambar 4.9. Keluaran sistem dengan pengendali MPC constrained ($Q = I_3, R = 100 I_6$)



Gambar 4.10. Sinyal kendali pengendali MPC constrained

$$(Q = I_3, R = 100 I_6)$$

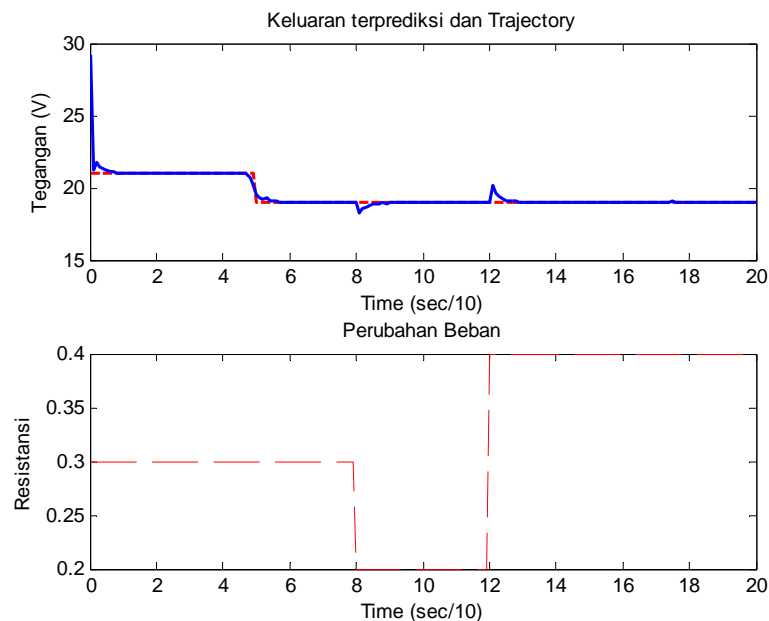
Jika nilai matriks bobot sinyal kendali $Q = I_3$ dan $R = 100 I_6$, sinyal kendali menjadi tidak halus lagi, dengan perubahan sinyal yang menjadi tidak proporsional. Hal ini menyebabkan keluaran prediksi yang dihasilkan akan sangat sensitif terhadap gangguan dan perubahansinyal acuan yang terjadi.

Dari pengujian yang dilakukan, terlihat bahwa meskipun dengan merubah nilai matriks R , keluaran prediksi tetap dapat mengikuti sinyal acuan yang di berikan serta pengendali dapat mengatasi gangguan yang terjadi, sehingga keluaran dengan mudah dapat kembali mengikuti sinyal acuan. Dengan makin besarnya nilai R , maka nilai $\Delta u(k)$ semakin ditekan sehingga dihasilkan keluaran yang lebih halus. Namun rentang antara matriks R dan Q yang terlalu besar menghasilkan sinyal kendali yang tidak baik sehingga pada nilai matriks $R = 100 I_6$, keluaran sistem tidak sehalus keluaran dengan nilai matriks R yang lebih kecil. Pada pengujian selanjutnya, nilai matriks R yang digunakan adalah $10 I_6$.

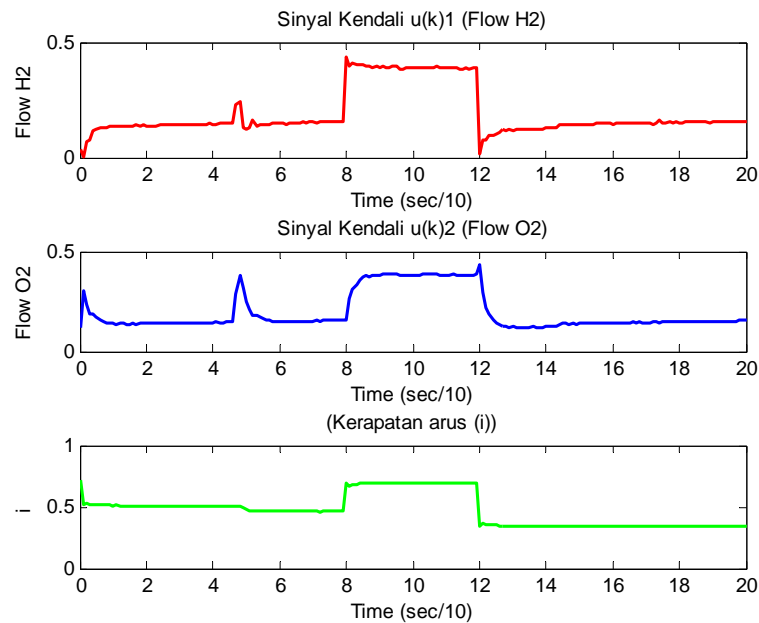
Semua sinyal kendali yang dihasilkan selalu memenuhi constraint (batasan) maksimum dan minimum untuk $u(k)$, namun untuk persamaan $u_2(k) = 0.89 u_1(k)$, tidak semua sinyal kendali yang dihasilkan dapat memenuhi batasan tersebut. Inilah yang disebut solusi *infeasible* dalam sistem pengendali MPC. Kondisi ini, membuat MPC mencari solusi lain untuk menentukan sinyal kendali yang bagaimana yang dikeluarkan agar tetap mempertahankan keluaran prediksi sesuai dengan sinyal acuan (minimasi error). Umumnya, hal ini terjadi jika terdapat perubahan sinyal acuan dan saat terjadinya gangguan. Pada sistem PEMFC sendiri, hal ini tidak menjadi masalah selama sinyal kendali tetap berada dalam batasan maksimum minimum yang ditetapkan.

4.2.1 Perbandingan hasil pengendalian MPC Constrained dengan Matriks Bobot Q yang berubah-ubah

a. $Q = 0.1 I_3, R = 10I_6$



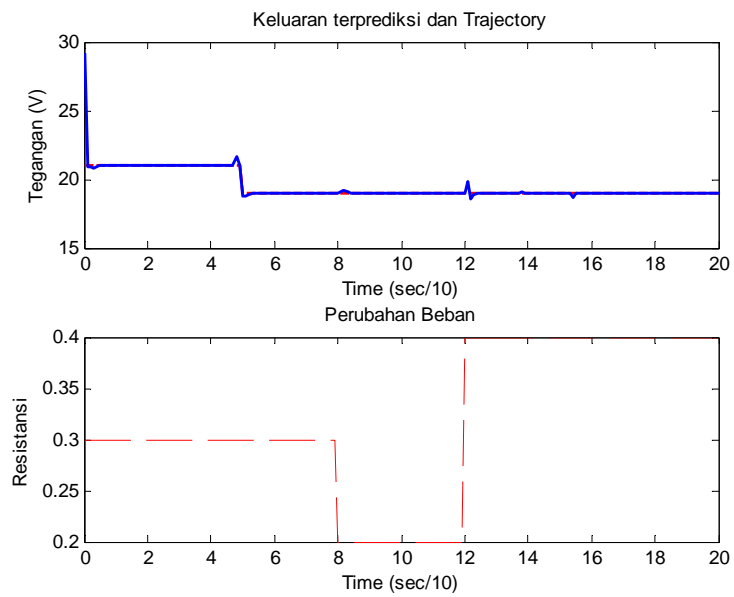
Gambar 4.11. Keluaran sistem dengan pengendali MPC constrained ($Q=0.1I_3, R=10I_6$)



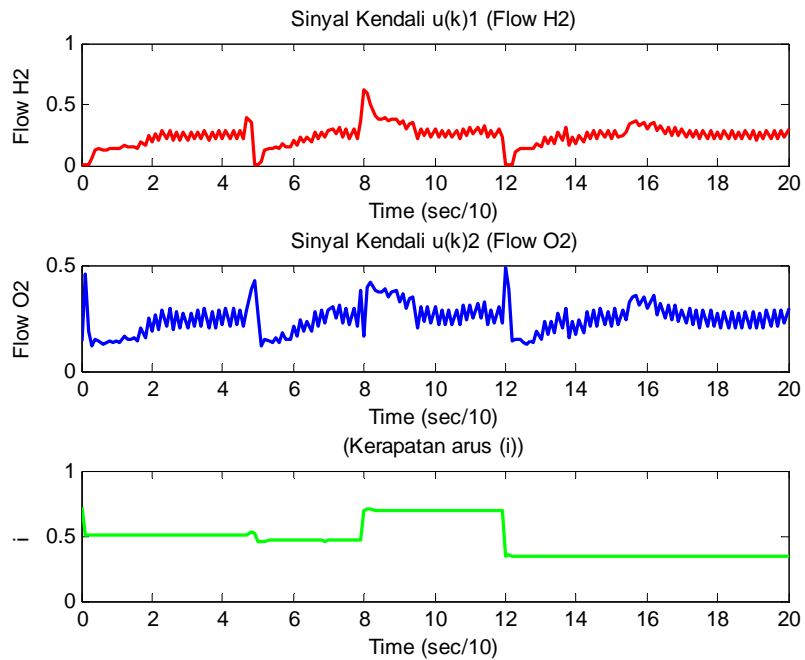
Gambar 4.12. Sinyal kendali pengendali MPC constrained ($Q=0.1I_3$, $R=10I_6$)

Untuk analisa terhadap pengaruh perubahan matriks bobot kesalahan Q , pengujian dilakukan dari $Q=0.1I_3$, $R=10I_6$. Pada uji ini, sinyal kendali yang dihasilkan cukup halus, namun sinyal keluaran sensitif terhadap pengaruh perubahan sinyal acuan maupun gangguan yang terjadi. Dalam hal ini, dapat dikatakan bahwa error yang terjadi tidak dapat diatasi dengan lebih cepat oleh pengendali karena bobot kesalahannya diperkecil. Pada uji ini juga masih terdapat solusi *infeasible* yaitu pada saat terjadi perubahan sinyal acuan di titik 5 detik dan saat terjadi perubahan beban di titik 12 detik.

b. $Q = 10 I_3, R = 10I_6$



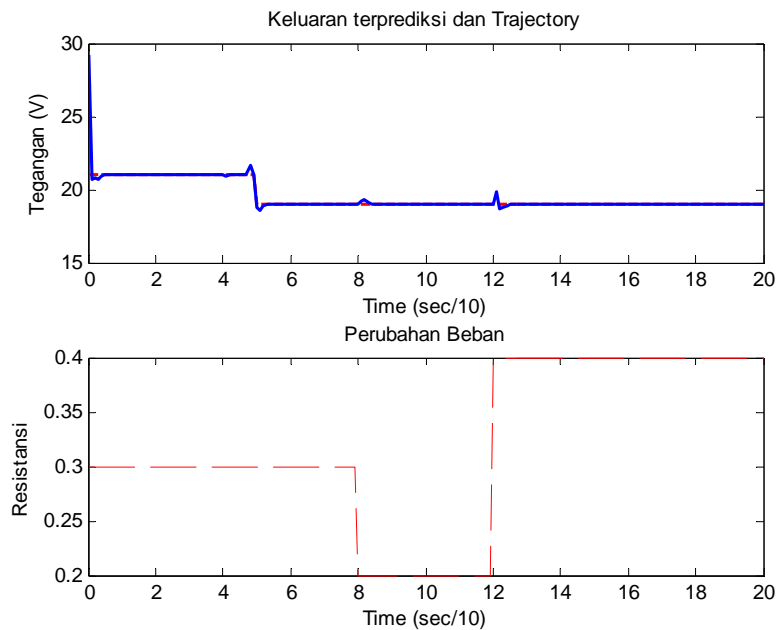
Gambar 4.13. Keluaran sistem dengan pengendali MPC constrained
($Q=10I_3, R=10I_6$)



Gambar 4.14. Sinyal kendali pengendali MPC constrained
($Q=10I_3, R=10I_6$)

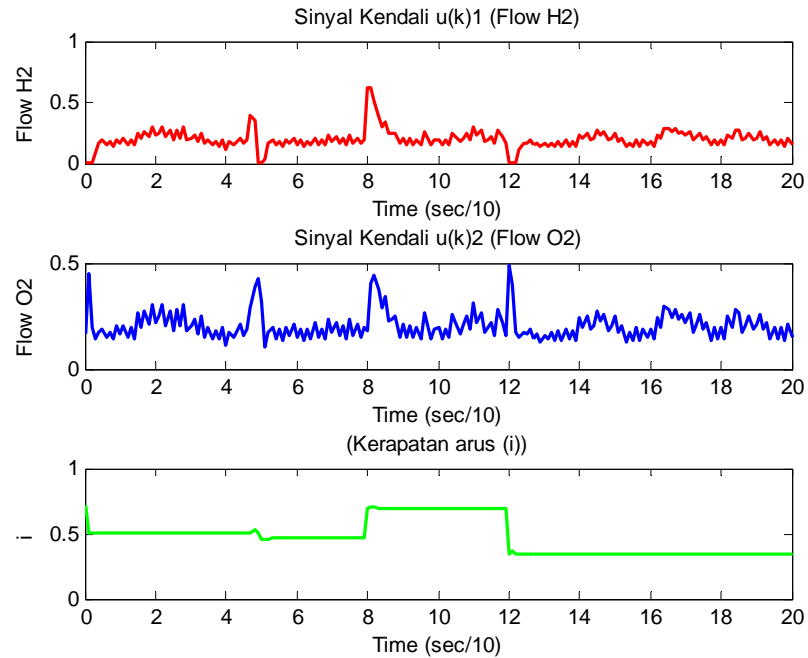
Dengan memperbesar bobot matriks Q menjadi $10I_3$, keluaran yang dihasilkan sangat halus, mampu mengikuti perubahan sinyal acuan dengan sangat baik dan pengaruh gangguan dapat diatasi dengan sangat baik pula. Namun, karena bobot matriks kesalahan yang diperbesar, maka perubahan sinyal kendali tidak ditekan untuk mengatasi error yang terjadi dengan cepat, sehingga sinyal kendali menjadi tidak halus, dan solusi *infeasible* tampak semakin banyak yaitu di titik 0.1, 5, 8 dan 12 detik.

c. $Q = 100I_3$, $R = 10I_6$



Gambar 4.15. Keluaran sistem dengan pengendali MPC constrained

$(Q=100I_3, R=10I_6)$



Gambar 4.16. Sinyal kendali pengendali MPC constrained
 $(Q=100I_3, R=10I_6)$

Dengan merubah besar nilai matriks Q dari nilai kecil hingga besar, terlihat bahwa untuk sistem PEMFC, untuk nilai paling besar (100), sinyal kendali berubah semakin bebas dan tidak di tekan.

Dari keseluruhan hasil percobaan, terlihat bahwa hasil pengendalian yang terbaik adalah dengan menggunakan matriks bobot kesalahan $Q = 1 I_3$ dan matriks bobot sinyal kendali $R = 10I_6$ untuk panjang *prediction horizon* dan *control horizon* = 3 seperti pada gambar (4.7 dan 4.8), dimana perubahan sinyal kendali halus dan keluaran prediksi paling mendekati sinyal acuan.

BAB 5

KESIMPULAN

Berdasarkan hasil pengujian pengendali Model Predictive Control yang telah dilakukan, dapat disimpulkan beberapa hal, yaitu:

1. Untuk sistem PEMFC, pengendali MPC yang di gunakan harus diberikan batasan (*constraint*) agar sinyal kendali yang dihasilkan sesuai dengan karakteristik fisik dari sistem PEMFC.
2. Dengan model yang baik, algoritma MPC dapat bekerja dengan baik yaitu mampu menghasilkan keluaran prediksi yang dapat mengikuti perubahan sinyal acuan dan mampu mengatasi gangguan (*disturbance*) yang terjadi.
3. Karena panjang rentang *prediction horizon* (H_p) dan *control horizon* (H_u) sama, maka parameter pengendali MPC dapat di atur pada matriks bobot Q dan R . Dari pengujian yang telah dilakukan, matriks bobot $Q = 1$ dan $R=10$
4. Adanya sinyal kendali yang tidak memenuhi constraint yang diinginkan, menunjukkan adanya solusi yang *infeasible*. Meskipun demikian, solusi tersebut tidak mempengaruhi kinerja pengendali MPC yang dirancang.

DAFTAR REFERENSI

U.S. Department of Energy Office of Fossil Energy , EG&G Technical Services, Inc, *Fuel Cell Handbook* (seventh edition) (National Energy Technology Laboratory, Morgantown, November 2004)

Subiantoro, Aries, *Diktat Kuliah Sistem Kendali Adaptif* (Depok : Control System Research Group Jurusan Elektro FTUI, 2002)

Larminie, James & Dicks, Andrew, *Fuel Cell System Explained* (John Wiley and Sons, February 2001)

Na, Woonki & Gou, Bei and Diong, Bill, Nonlinear Control of PEM Fuel Cell by Exact Linearization, IEEE 2005