

Kontrol Optimal pada *Balancing Robot* Menggunakan Metode *Linear Quadratic Regulator*

Juliana. Sumanti, Arie S. M. Lumenta, ST, MT, David Pang, ST, MT,
Jurusan Teknik Elektro-FT, UNSRAT, Manado-95115, Email: juliana.sumanti@gmail.com

Abstrak - *Balancing robot is a mobile robot that has wheels on its left and right sides which construction is unstable and must be actively balanced to stay upright. Balancing robot is a development from the reverse-pendulum model which is then put on wheels. To control the balancing robot's system, Linear Quadratic Regulator (LQR) method is used as a feedback, which is simulated using MATLAB's simulink program.*

The analysis and simulations yields optimum results at weight matrix values $Q(1,1) = 100$, $Q(3,3) = 100$, and $R = 10$. System's cart settling time is at 3.8 seconds with stability at 0.063 meter and 4 seconds for the pendulum. Delay time for the cart is at 1.8 seconds.

Keywords: *Balancing Robot, MATLAB, Optimal, Simulink, Time*

Abstrak - *Balancing robot merupakan suatu robot mobile yang memiliki roda disisi kanan dan kirinya yang tidak stabil dan harus diseimbangkan secara aktif untuk menjaganya tetap tegak. Balancing robot merupakan pengembangan dari model pendulum terbalik yang diletakkan diatas kereta beroda. Untuk mengontrol sistem balancing robot digunakan metode kontrol Linear Quadratic Regulator (LQR) sebagai umpan balik yang disimulasikan menggunakan program simulink MATLAB.*

Dari hasil analisa dan simulasi diperoleh kesimpulan hasil yang paling optimal pada nilai matriks bobot $Q(1,1) = 100$, $Q(3,3) = 100$ dan $R = 10$. Respon sistem mencapai keadaan optimal (settling time) untuk cart 3.8 detik dengan kestabilan pada jarak 0.063 meter dan untuk pendulum 4 detik. Waktu tunda (delay time) untuk cart 1.8 detik.

Kata Kunci: *Balancing Robot, MATLAB, Optimal, Simulink, Waktu*

I. PENDAHULUAN

Perkembangan teknologi robotika telah membuat kualitas kehidupan manusia semakin tinggi. Saat ini perkembangan teknologi robotika telah mampu meningkatkan kualitas maupun kuantitas berbagai industri. Teknologi robotika juga telah menjangkau sisi hiburan dan pendidikan bagi manusia. salah satu cara menambah tingkat kecerdasan suatu robot yaitu dengan menambah metode kontrol bahkan memberikan kecerdasan buatan pada robot tersebut. Salah satunya adalah *balancing robot*.

Balancing robot (robot penyeimbang) merupakan suatu robot *mobile* yang memiliki roda di sisi kanan dan kirinya yang tidak akan seimbang tanpa adanya kontroler. *Balancing robot* ini merupakan pengembangan dari model pendulum terbalik (*inverted pendulum*) yang diletakkan diatas kereta beroda. Menyeimbangkan *balancing robot* memerlukan suatu metode kontrol yang baik untuk mempertahankan posisi robot dalam keadaan tegak lurus. Bahkan sekarang ini, konsep

balancing robot telah digunakan sebagai alat transportasi yang bernama *segway*.

Dalam Tugas Akhir ini akan dikaji masalah pengendalian sistem *balancing robot* dengan menggunakan kontrol optimal *Linear Quadratic Regulator* (LQR) yang kemudian akan di simulasikan ke dalam *simulink* MATLAB.

II. LANDASAN TEORI

A. Dasar Sistem Kendali

Sistem adalah kumpulan objek yang saling berinteraksi dan bekerja sama untuk mencapai tujuan logis dalam suatu lingkungan yang kompleks. Sistem kendali adalah kumpulan obSistem kendali adalah kumpulan komponen yang bekerja sama di bawah arahan dari sebuah atau beberapa mesin cerdas (*machine intelligence*). Di samping itu, sistem kendali juga dapat diartikan sebagai proses pengaturan/pengendalian terhadap satu atau beberapa besaran (variabel, parameter) sehingga berada pada suatu harga atau dalam suatu rangkuman harga (*range*) tertentu.

Sistem kendali terdiri dari dua model sistem dasar, yakni sistem kendali loop terbuka dengan sistem kendali loop tertutup.

Sistem Kendali Loop Terbuka

Sistem kendali loop terbuka adalah sistem kendali yang keluarannya tidak terpengaruh pada aksi pengendalian. Jadi pada sistem kendali loop terbuka, keluaran tidak diukur atau diumpan-balikkan untuk dibandingkan dengan masukan. Gambar 1 menunjukkan hubungan masukan dan keluaran untuk sistem kendali loop terbuka.

Setiap sistem kendali loop terbuka, keluaran tidak dibandingkan dengan masukan. Sehingga untuk setiap masukan, terdapat satu kondisi operasi yang tetap. Jadi, ketelitian sistem bergantung pada kalibrasi. Apabila terjadi gangguan, sistem kendali loop terbuka tidak dapat bekerja seperti yang diinginkan. Kendali loop terbuka dapat digunakan dalam praktek hanya jika hubungan antara masukan keluaran diketahui dan tidak terdapat gangguan internal maupun eksternal.

Sistem Kendali Loop Tertutup

Sistem kendali loop tertutup adalah sistem kendali yang sinyal keluarannya mempunyai pengaruh langsung pada aksi pengendalian. Jadi, sistem kendali loop tertutup adalah sistem kendali yang berumpun-balik.

Sinyal kesalahan yang merupakan selisih antara sinyal masukan dan sinyal umpan-balik, diumpun ke kontroler untuk memperkecil kesalahan dan membuat agar keluaran sistem

mendekati harga yang diinginkan. Gambar 2 menunjukkan hubungan masukan-keluaran dari sistem kendali loop tertutup.

B Persamaan Ruang Keadaan

Persamaan ruang keadaan (*state-space equation*) dari sistem dinamik mengandung tiga hal, yaitu variabel input (*input variable*), variabel output (*output variable*) dan variabel keadaan (*state variable*).

Persamaan ruang keadaan dari suatu sistem dapat bervariasi, sesuai dengan definisi awal dari variabel-variabel dari suatu sistem. Misalkan suatu sistem memiliki *state* sejumlah *n* (persamaan diferensial biasa berdimensi *n*), input sebanyak *r*, dan output sebanyak *m*. Misalkan pula $x = (x_1, x_2, \dots, x_n)$, $u = u_1, u_2, \dots, u_r$. Maka, sistem tersebut dapat dituliskan sebagai berikut:

$$\begin{aligned} \dot{x}_1 &= f_1(x, u, t), \\ \dot{x}_2 &= f_2(x, u, t), \\ &\vdots \\ \dot{x}_n &= f_n(x, u, t) \end{aligned} \tag{1}$$

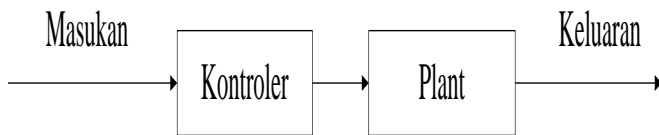
Sedangkan output dari sistem diberikan sebagai berikut:

$$\begin{aligned} y_1 &= g_1(x, u, t) \\ y_2 &= g_2(x, u, t) \\ &\vdots \\ y_n &= g_n(x, u, t) \end{aligned} \tag{2}$$

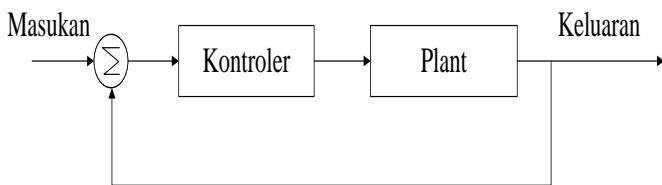
Persamaan (1) dan (2) dapat dituliskan dalam notasi vektor sebagai berikut:

$$\dot{x}(t) = f(x, u, t) \tag{3}$$

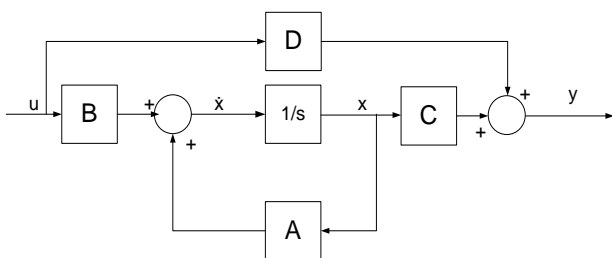
$$y(t) = g(x, u, t) \tag{4}$$



Gambar 1. Komponen Sistem Kendali Loop Terbuka



Gambar 2. Komponen Sistem Kendali Loop Tertutup



Gambar 3. Blok Diagram Persamaan Keadaan (*State Space*)

Dengan:

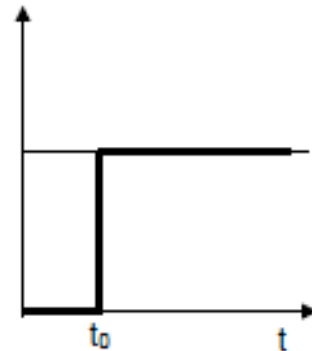
$$\begin{aligned} x &= \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}, y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, u = \begin{bmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{bmatrix} \\ f(x, u, t) &= \begin{bmatrix} f_1(x, u, t) \\ f_2(x, u, t) \\ \vdots \\ f_n(x, u, t) \end{bmatrix} \\ g(x, u, t) &= \begin{bmatrix} g_1(x, u, t) \\ g_2(x, u, t) \\ \vdots \\ g_n(x, u, t) \end{bmatrix} \end{aligned}$$

Jika vektor fungsi *f*, *g* bergantung kepada variabel *t*, maka persamaan (3) dan (4) disebut sistem *time-varying*. Jika sistem tersebut dilinearkan, maka persamaan linear ruang keadaan dalam persamaan outputnya dapat dituliskan sebagai berikut:

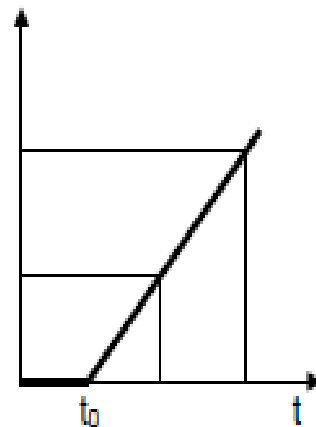
$$\dot{x}(t) = A(t)x(t) + B(t)u(t) \tag{5}$$

$$y(t) = C(t)x(t) + D(t)u(t) \tag{6}$$

Dengan *A(t)* disebut matriks keadaan, *B(t)* matriks masukan, *C(t)* matriks keluaran, dan *D(t)* matriks transmisi langsung.



Gambar 4. Grafik Fungsi Step



Gambar 5. Grafik Fungsi Ramp

Bila fungsi vektor f dan g tidak bergantung terhadap waktu t , maka sistem disebut sistem *time-invariant*. Dalam hal ini, Persamaan (3) dan (4) dapat disederhanakan menjadi:

$$\dot{x}(t) = f(x, u) \quad (7)$$

$$y(t) = g(x, u) \quad (8)$$

Persamaan (7) dan (8) dapat dilinearkan di sekitar kedudukan operasi sebagai berikut:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (9)$$

$$y(t) = Cx(t) + Du(t) \quad (10)$$

Diagram blok yang mewakili persamaan (9) dan (10) ditunjukkan dalam gambar 3.

C Karakteristik Respon

Karakteristik respon adalah ciri-ciri khusus perilaku respon dinamik (spesifikasi performansi) output sistem yang muncul akibat diberikannya suatu sinyal masukan tertentu yang khas bentuknya (disebut sebagai sinyal uji).

Sinyal uji merupakan sinyal masukan uji (*test input signal*) yang biasa digunakan adalah fungsi tangga, fungsi “ramp”, fungsi percepatan, fungsi impuls, fungsi sinusoida dan sebagainya. Dengan sinyal uji ini dapat dilakukan analisis matematika dan eksperimental sistem kontrol secara mudah karena sinyal-sinyal ini merupakan fungsi waktu yang sangat sederhana. Jenis sinyal masukan yang akan digunakan untuk menganalisis karakteristik sistem di antara sinyal-sinyal masukan khas ini, dapat ditentukan dari bentuk masukan yang paling sering akan diberikan ke sistem pada operasi normal. Jika masukan sistem kontrol merupakan fungsi waktu yang berangsur-angsur berubah, maka fungsi waktu “ramp” mungkin merupakan sinyal uji yang baik. Demikian pula jika sistem dikenai gangguan secara tiba-tiba, maka fungsi waktu berjenjang (*step*) mungkin merupakan sinyal uji yang baik. Dan untuk sistem yang dikenai masukan-masukan kejut, sinyal uji yang paling baik mungkin adalah fungsi impuls. Setelah sistem kontrol di disain berdasarkan sinyal uji, kinerja sistem dalam memberikan respon terhadap masukan yang sebenarnya biasanya memuaskan. Penggunaan sinyal uji memungkinkan kita untuk membandingkan kinerja semua sistem dengan basis yang sama. Gambar 4 dan Gambar 5 memberikan gambaran contoh sinyal uji fungsi *step* dan fungsi *ramp*.

Keluaran yang dihasilkan merupakan (*response*) dari sistem yang diberikan sinyal uji. Bila analisa yang dilakukan merupakan analisa dalam lingkup waktu dan masukan yang diberikan bukan merupakan fungsi periodik (mempunyai frekuensi), maka analisa tersebut merupakan analisa tanggapan waktu (*time response*).

Karakteristik Respon Waktu (Time Respons)

Karakteristik respon waktu adalah karakteristik respon yang spesifikasi performansinya didasarkan pada pengamatan bentuk respon output sistem terhadap berubahnya waktu. Secara umum spesifikasi performansi respon waktu dapat dibagi atas dua tahapan pengamatan yaitu spesifikasi respon transien dan spesifikasi respon *steady state*.

Spesifikasi Respon Transien Spesifikasi respon transien adalah spesifikasi respon sistem yang diamati mulai saat

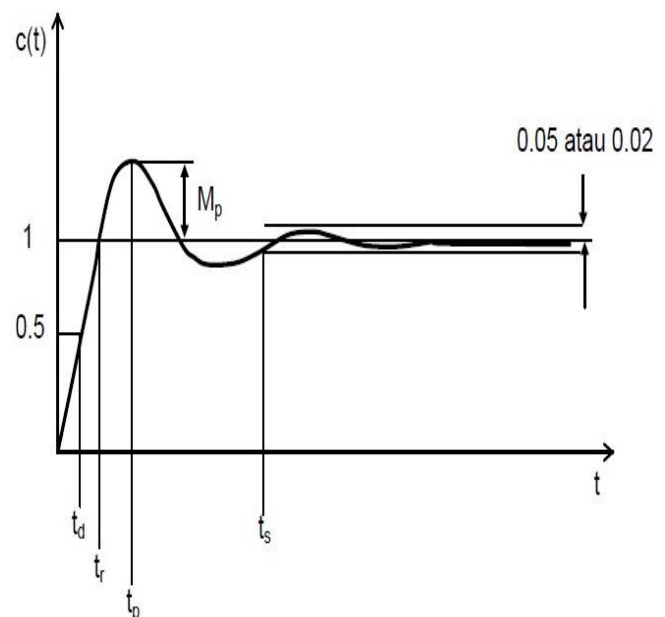
terjadinya perubahan sinyal input/gangguan/beban sampai respon masuk dalam keadaan *steady state*. Tolak ukur yang digunakan untuk mengukur kualitas respon transien secara grafik ditunjukkan pada gambar 6 dan didefinisikan dalam lima hal yaitu waktu tunda, waktu naik, waktu puncak, *overshoot* maksimum dan waktu turun. Waktu tunda (t_d) adalah waktu yang diperlukan oleh tanggapan untuk mencapai setengah nilai akhir untuk waktu yang pertama. Waktu naik (t_r) adalah waktu yang diperlukan oleh tanggapan untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0% menjadi 100% dari nilai akhir yang biasa digunakan. Untuk sistem atas redaman waktu naik yang biasa digunakan 10% menjadi 90%. Waktu puncak (t_p) adalah waktu yang diperlukan untuk mencapai puncak pertama *overshoot*. *Overshoot* maksimum (M_p) adalah nilai puncak kurva tanggapan diukur dari satuan. Apabila nilai akhir keadaan tunak tanggapannya jauh dari satu, maka biasa digunakan persen *Overshoot* maksimum, dan didefinisikan oleh

$$\text{Maksimum (persen) Overshoot} =$$

$$\frac{c(t_p) - c(\infty)}{c(\infty)} \times 100\%$$

Besarnya persen *Overshoot* maksimum menunjukkan kestabilan relatif dari sistem. Waktu turun (t_s) adalah waktu yang diperlukan untuk menanggapi kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang disederhanakan dengan persentase mutlak harga akhir (2% atau 5%). Waktu turun tadi dihubungkan dengan tetapan waktu terbesar sistem kontrol. Apabila kita menemukan kriteria kesalahan persentase untuk sistem kontrol, kita boleh menetapkannya dari tujuan desain sistem dalam pertanyaan.

Spesifikasi respon *steady state* adalah spesifikasi respon sistem yang diamati mulai saat respon masuk dalam keadaan *steady state* sampai waktu tak terbatas. Tolak ukur yang digunakan untuk mengukur kualitas respon *steady state* ini antara lain *error steady state* baik untuk *error* posisi, *error* kecepatan maupun *error* percepatan.



Gambar 6. Penggolongan Tanggapan Transien

D Pendulum Terbalik (Inverted Pendulum)

Pendulum terbalik adalah sebuah pendulum (bandul) dimana pusat massanya berada di atas titik tumpunya. Seringkali diimplementasikan pada sebuah kereta yang dapat bergerak secara horizontal dan bias disebut sebagai sebuah kereta-galah. Kebanyakan aplikasi membatasi pendulum pada satu derajat kebebasan dengan mengasifikasi tiang (galah) pada sebuah sumbu putar. Dimana sebuah pendulum normal stabil ketika digantung menghadap ke bawah, sebuah pendulum terbalik bersifat tidak stabil dan harus diseimbangkan secara aktif untuk menjaganya tetap tegak. Ini bisa dilakukan dengan mengaplikasikan torsi pada titik tumpu, dengan menggerakkan titik tumpu secara horizontal sebagai bagian dari sistem umpan balik, merubah nilai putaran dari massa yang bertumpu pada pendulum dalam satu sumbu parallel ke sumbu tumpuan lalu setelah itu menciptakan nilai torsi pada pendulum, atau dengan mengosilasi titik tumpu secara vertikal. Demo sederhana dari menggerakkan titik tumpu dalam suatu system umpan balik bisa dilakukan dengan menyeimbangkan sapu terbalik pada ujung jari. Pendulum terbalik merupakan masalah yang klasik dalam dinamika dan teori control dan digunakan sebagai standar untuk mengukur suatu percobaan strategi kontrol.

Tipe kedua dari pendulum terbalik adalah sebuah pengukur kemiringan sebuah struktur bangunan yang tinggi, terdiri dari sebuah kabel yang ditanamkan pada dasar pondasi kemudian dikaitkan pada sebuah pelampung dalam sebuah kolam minyak di atas struktur yang memiliki alat pengukur perpindahan posisi normal pelampung kemudian dibandingkan dengan posisi sebenarnya.

E Konsep Sistem Kontrol Optimal Menggunakan Metode Linear Quadratic Regulator (LQR)

Sistem optimal adalah sistem yang mempunyai unjuk kerja terbaik (*best performance*) terhadap suatu acuan tertentu. Sistem kontrol optimal memerlukan adanya suatu kriteria optimasi yang dapat meminimumkan hasil pengukuran dengan deviasi perilaku sistem terhadap perilaku idealnya.

Pengukuran tersebut dilakukan dengan menentukan indeks performansi yang merupakan suatu fungsi dari suatu harga yang dapat dianggap menunjukkan seberapa besar kinerja sistem yang sesungguhnya sesuai dengan kinerja yang diinginkan. Indeks performansi merupakan tolak ukur suatu sistem optimal. Sistem akan optimal bila indeks performansinya adalah minimum.

Agar sistem tersebut dapat dikontrol, maka perlu dibuat model matematis yang menghubungkan antara masukan (*input*) dan keluaran (*output*). Pada sistem kontrol optimal, model yang banyak digunakan adalah persamaan keadaan (*state space*).

Teori Regulator Optimal

Dalam beberapa proses, variabel yang dikontrol akan mengalami deviasi karena adanya gangguan. Regulator kontrol dirancang untuk melakukan kompensasi terhadap gangguan.

Linear Quadratic Control merupakan salah satu metode dalam perancangan sistem kontrol optimal. *Plant* diasumsikan bersifat sistem linear, dalam bentuk persamaan keadaan, dan

fungsi obyektif adalah fungsi kuadratik dari keadaan *plant* dan sinyal input kendali.

Kelebihan penggunaan formulasi *Linear Quadratic* adalah pada kemudahan analisa dan pengimplementasiannya. Beberapa masalah yang biasa diselesaikan dengan metode ini adalah masalah minimasi waktu, dan lain-lain.

Linear Quadratic Regulator (LQR)

Metode optimasi dengan *linear quadratic regulator* adalah dengan menentukan sinyal masukan yang akan memindahkan suatu *state* sistem linear dari kondisi $x(t_0)$ menuju ke suatu kondisi akhir $x(t)$ yang akan meminimumkan suatu indeks untuk kerja performansi kuadratik. *Cost function* yang dimaksud adalah waktu integral dari bentuk kustratis pada vektor keadaan (*state*) x dan u seperti pada persamaan:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \tag{11}$$

Dimana Q adalah faktor pembobotan *state* (matriks semi definit positif) dan R adalah bobot faktor variable kontrol (matriks definit positif). Dengan persamaan seperti diatas, variasi parameter dari masalah perancangan LQR dapat ditentukan, juga untuk kondisi akhir yang mungkin dapat berpengaruh pada *cost function*.

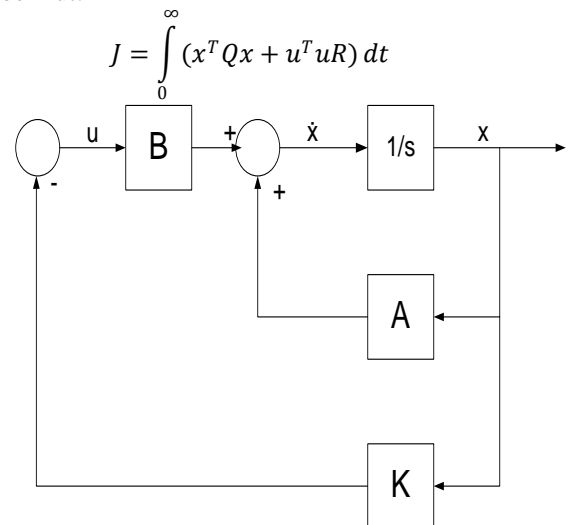
Prinsip penggunaan metode LQR adalah memperoleh sinyal kendali optimal sinyal kendali optimal dari umpan balik keadaan (*state feedback*).

$$u = -Kx \tag{12}$$

Matriks umpan balik K diperoleh dengan memecahkan persamaan Riccati. Salah satu kendala penggunaan metode LQR adalah pemecahan persamaan Riccati yang tidak mudah jika diselesaikan secara manual, maka dibutuhkan bantuan komputer, dalam hal ini dengan paket program MATLAB.

Controller Algebraic Riccati Equation (CARE)

Untuk sistem linier, *time-invariant*, dapat dirutunkan persamaan Aljabar Riccati untuk mencari solusi optimal sebagai berikut:



Gambar 7. Blok Digram LQR

Formulasi dan solusi masalah LQR untuk waktu *continuous*, dengan umpan balik keadaan dituliskan sebagai berikut:

$$\begin{aligned} u &= -Kx \\ K &= R^{-1}B^T P \end{aligned} \quad (13)$$

Syarat cukup untuk kontrol optimal matriks P harus memenuhi:

$$\frac{dP}{dt} = A^T P + PA - PBR^{-1}B^T P + Q = 0 \quad (14)$$

Persamaan (14) dikenal sebagai Persamaan Riccati (*Riccati Equation*). Dengan syarat matriks A dan B, *controllable* dan *observable*. Blok diagram sistem kontrol optimal dengan umpan balik keadaan dapat dilihat pada gambar 7.

III. METODE PENELITIAN

A Rencana Penelitian

Untuk merealisasikan tujuan yang ingin dicapai dalam penulisan skripsi ini, langkah-langkah yang akan dilakukan adalah sebagai berikut:

Bahan Penelitian

Data yang dikumpulkan diperoleh dari studi literatur dan bentuk *real* mekanik *balancing* robot dari skripsi yang disusun oleh Samuel Yosia Dimpudus yang berjudul "PERANCANGAN SISTEM PENGATURAN *BALANCING* ROBOT". Data-data tersebut berupa parameter-parameter, dan komponen yang menyusun sistem.

Pemodelan Matematik Sistem

Langkah-langkah yang dilakukan untuk mendapatkan model matematis sistem antara lain menentukan sistem yang akan diteliti dan komponen-komponen yang menyusun sistem tersebut. Setelah diketahui komponen-komponen penyusun sistem beserta parameter-parameternya, maka sistem dapat dimodelkan menjadi sebuah persamaan matematis dalam bentuk persamaan keadaan (*state space*).

Analisis Sistem

Analisis optimal sistem bertujuan untuk mendapatkan *gain* kontrol untuk pengendalian optimal.

Untuk membuat sistem dapat teranalisa, maka di buat beberapa tahap. Pertama sistem dibuat dalam bentuk persamaan keadaan dilakukan pemeriksaan apakah sistem memenuhi syarat keteramatan dan kekontrolan. Jika tidak, maka berarti sistem tidak dapat dikontrol. Jika ya, maka proses dapat dilanjutkan. Kedua, tentukan matriks bobot Q dan R. Penentuan matriks bobot Q dan R ini dilakukan dengan cara coba-coba (*trial and error*) untuk mendapatkan matriks umpan balik LQR. Ketiga, analisa hasil sistem apakah sudah memenuhi dua criteria antara yaitu waktu stabil (*settling time*) untuk *cart* dan pendulum kurang dari 5 detik dan waktu tunda (*delamy time*) untuk *cart* kurang dari 2 detik. Jika sistem tidak memnuhi dua criteria tersebut, atur kembali matriks bobot Q dan R. Jika ya, maka proses selesai.

Simulasi Sistem

Simulasi sistem dilakukan dengan menggunakan fasilitas-fasilitas yang tersedia pada perangkat lunak program

MATLAB, antara lain *simulink* dan *m-file*. Dari hasil simulasi tersebut, dapat dibandingkan respon sistem terhadap pengaruh pengontrol LQR dalam bentuk sinyal keluaran.

B Cara Kerja Penelitian

Dalam usaha untuk mencapai tujuan dan menjawab rumusan masalah dalam skripsi ini, dilakukan tiga tahap pengerjaan yaitu studi literatur untuk menganalisa masalah kontrol optimal pada *plant*, dilakukan dengan menghimpun informasi dari buku-buku dan jurnal-jurnal yang berkaitan. Buku dan jurnal-jurnal tersebut sebagian diperoleh dari Laboratorium Teknik Kendali, Jurusan Teknik Elektro Universitas Sam Ratulangi dan beberapa di antaranya di peroleh dari internet. Membuat dan menganalisa hasil simulasi sistem. Setelah system telah dianalisa, maka dapat ditarik kesimpulan.

C Alat Penelitian

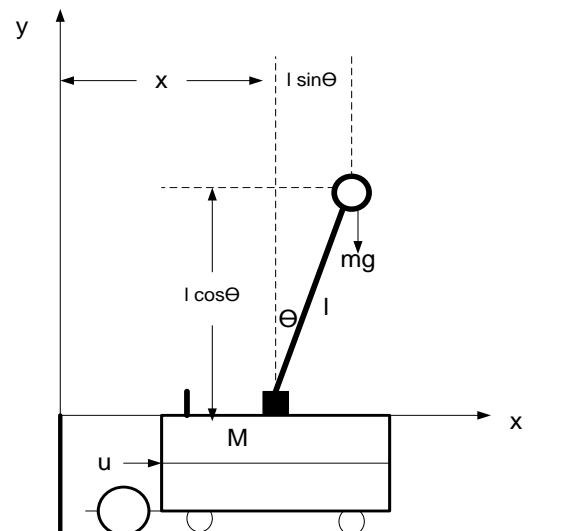
Penelitian ini menggunakan perangkat keras (*hardware*) berupa Laptop dengan spesifikasi sebagai berikut Processor : Intel(R) Core(TM) i3-2367M CPU @ 1.40GHz 1.40 GHz. Installed memory (RAM) : 4.00 GB (3.06 GB usable). System type : 32-bit Operating System

Untuk perangkat lunak (*software*) yaitu MATLAB dengan spesifikasi sebagai berikut Version 7.11.0.584 (R2010b). 32-bit (win32)

D Model Sistem Balancing Robot

Balancing robot merupakan suatu robot *mobile* yang memiliki roda disisi kanan dan kirinya yang tidak akan seimbang apabila tanpa adanya pengendali. *Balancing* robot ini merupakan pengembangan dari model pendulum terbalik yang diletakkan diatas kereta beroda.

Gambar 8 adalah sebuah contoh dari *balancing* robot. Dalam kasus ini, kereta yang dilengkapi motor hanya dapat bergerak garis lurus (*horizontal*), dan pendulum yang diletakkan di atas kereta bergerak (*berotasi*) dalam bidang yang sama. Gaya $u(t)$ diberikan kepada kereta. Tanda adanya gaya yang sesuai, pendulum akan jatuh. Dengan adanya umpan balik, motor pada kereta akan memberikan gaya yang sesuai sehingga pendulum tetap dalam keadaan tegak.



Gambar 8. Model Dari *Balancing* Robot (Tampak Samping)

Dimana:

θ : sudut antara pendulum dengan garis vertikal.

M : berat kereta (kg).

m : berat pendulum (kg).

$\dot{\theta}$: kecepatan sudut $\left(\frac{d\theta}{dt}\right)$.

$\ddot{\theta}$: percepatan sudut $\left(\frac{d^2\theta}{dt^2}\right)$.

\dot{x} : kecepatan benda $\left(\frac{dx}{dt}\right)$.

\ddot{x} : percepatan benda $\left(\frac{d^2x}{dt^2}\right)$.

I : momen inersia.

b : koefisien dari gaya gesek (*viscount friction*) antara kereta dengan lantai.

V : gaya reaksi vertikal pada pendulum.

H : gaya reaksi horisontal pada pendulum.

u : gaya input yang diberikan pada kereta (N).

Asumsi

Berikut adalah asumsi-asumsi dalam memodelkan *balancing* robot yaitu pendulum homogen (rapat massa di setiap titik pada pendulum sama), sehingga I (momen inersia) $= \frac{1}{3}ml^2$ dan θ kecil ($\cos \theta \approx 1$ dan $\sin \theta \approx \theta$).

Formulasi model

Berikut ini akan diturunkan model matematik untuk sistem *balancing* robot. Setelah mendapatkan model matematik sistem, akan dilakukan pengendalian terhadap sistem *balancing* robot.

Penjumlahan gaya dari *cart* secara horisontal dirumuskan dalam persamaan berikut:

$$\begin{aligned} \sum F_H &= Ma \\ u - H - b\dot{x} &= M\ddot{x} \\ u &= M\ddot{x} + b\dot{x} + H^{[3]} \end{aligned} \quad (15)$$

Karena *cart* bergerak arah horisontal, maka penjumlahan gaya pada arah vertikal diabaikan. pada gambar 8 *free body diagram* dapat dibagi menjadi 2 yaitu *free body diagram* untuk *cart* dan *free body diagram* untuk pendulum. Dengan menjumlahkan gaya pada arah horisontal maka didapatkan persamaan gaya H.

$$H = m \frac{d^2}{dt^2} (x + l \sin \theta)$$

Dengan memperhatikan bahwa:

$$\frac{d^2}{dt^2} \sin \theta = -(\sin \theta) \dot{\theta}^2 + (\cos \theta) \ddot{\theta}^{[1]}$$

Maka persamaan menjadi:

$$H = m\ddot{x} - ml\dot{\theta}^2 \sin \theta + ml\ddot{\theta} \cos \theta \quad (16)$$

Dengan mensubstitusikan persamaan (16) ke persamaan (15) maka didapatkan persamaan gerak yaitu:

$$u = (M + m)\ddot{x} + b\dot{x} - ml\dot{\theta}^2 \sin \theta + ml\ddot{\theta} \cos \theta \quad (17)$$

Untuk mendapatkan persamaan gerak selanjutnya, maka gaya-gaya yang tegak lurus dengan batang pendulum dijumlahkan.

$$V \sin \theta + H \cos \theta - mg \sin \theta = ml\ddot{\theta} + m\ddot{x} \cos \theta \quad (18)$$

Untuk mengeliminasi V dan H dari persamaan (18), jumlahkan momen-momen di sekitar pendulum.

$$\begin{aligned} \sum \tau_{pend} &= I\ddot{\theta} \\ -Vl \sin \theta - Hl \cos \theta &= I\ddot{\theta} \end{aligned} \quad (19)$$

Substitusikan persamaan (19) ke persamaan (18) maka didapatkan:

$$\begin{aligned} -I\ddot{\theta} - mgl \sin \theta &= ml^2\ddot{\theta} + ml\ddot{x} \cos \theta \\ (I + ml^2)\ddot{\theta} &= -mgl \sin \theta - ml\ddot{x} \cos \theta \end{aligned} \quad (20)$$

Dapat dilihat bahwa persamaan (16) dan persamaan (20) adalah persamaan diferensial nonlinear. Untuk menjaga agar *balancing* robot tetap vertikal, maka dapat diasumsikan bahwa θ yang dihasilkan kecil sehingga $\cos \theta \approx 1$, $\sin \theta \approx \theta$, dan $\theta\dot{\theta}^2 \approx 0$. Kemudian persamaan (17) dan (20) dapat dilinearisasi sebagai berikut:

$$u = (M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta} \quad (21)$$

$$(I + ml^2)\ddot{\theta} = -mgl\theta - ml\ddot{x} \quad (22)$$

Persamaan Keadaan (State Space)

Misalkan vector

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix}$$

Dan

$$y = \begin{bmatrix} x \\ \theta \end{bmatrix} \text{ sebagai output dari sistem.}$$

Agar diperoleh persamaan state space linear untuk \ddot{x} , persamaan (21) harus merupakan fungsi dari turunan yang lebih rendah saja. Untuk itu, $\ddot{\theta}$ harus dieliminasi dari persamaan (21), dan diperoleh:

$$(M + m)\ddot{x} = u - b\dot{x} - ml \left(\frac{-mgl\theta - ml\ddot{x}}{(I + ml^2)} \right)$$

Atau

$$\begin{aligned} (M + m)\ddot{x} &= u - b\dot{x} + \frac{m^2 l^2 g}{(I + ml^2)} \theta + \frac{m^2 l^2}{(I + ml^2)} \ddot{x} \\ \ddot{x} &= \frac{(I + ml^2)u - b(I + ml^2)\dot{x} + m^2 l^2 g \theta}{I(M + m) + Mml^2} \\ \ddot{x} &= \frac{-b(I + ml^2)}{I(M + m) + Mml^2} \dot{x} + \frac{m^2 l^2 g}{I(M + m) + Mml^2} \theta + \frac{(I + ml^2)}{I(M + m) + Mml^2} u \end{aligned} \quad (23)$$

Persamaan berikut di peroleh dengan mengeliminasi \ddot{x} dari persamaan (22) untuk memperoleh persamaan *state space* linear untuk $\ddot{\theta}$.

$$\begin{aligned}
 (I + ml^2)\ddot{\theta} &= -mgl\theta - ml\left(\frac{u - b\dot{x} - ml\ddot{\theta}}{(M+m)}\right) \\
 (I + ml^2)\ddot{\theta} &= -mgl\theta - \frac{ml}{(M+m)}u + \frac{mlb}{(M+m)}\dot{x} + \frac{m^2l^2}{(M+m)}\ddot{\theta} \\
 \ddot{\theta} &= \frac{-(M+m)mgl\theta - mlu + mlb\dot{x}}{I(M+m) + Mml^2} \\
 \ddot{\theta} &= \frac{mlb}{I(M+m) + Mml^2}\dot{x} - \frac{(M+m)mgl}{I(M+m) + Mml^2}\theta - \frac{ml}{I(M+m) + Mml^2}u
 \end{aligned} \tag{24}$$

Berdasarkan pemisalan vektor x , dan dari persamaan (23) dan (24), diperoleh:

$$\begin{aligned}
 \dot{x}_1 &= x_2 = \dot{x} \\
 \dot{x}_2 &= \ddot{x} = \frac{-b(I+ml^2)}{I(M+m)+Mml^2}\dot{x} + \frac{m^2l^2g}{I(M+m)+Mml^2}\theta + \frac{(I+ml^2)}{I(M+m)+Mml^2}u \\
 \dot{x}_3 &= x_4 = \dot{\theta} \\
 \dot{x}_4 &= \ddot{\theta} = \frac{mlb}{I(M+m)+Mml^2}\dot{x} - \frac{(M+m)mgl}{I(M+m)+Mml^2}\theta - \frac{ml}{I(M+m)+Mml^2}u
 \end{aligned}$$

Dalam bentuk *state space*, persamaan matematika dari sistem dibuat ke dalam bentuk persamaan matriks berikut:

$$\begin{aligned}
 \dot{x} &= Ax + Bu \\
 y &= Cx + Du
 \end{aligned}$$

Dimana, A, B, C , dan D adalah matriks, sementara x adalah *state*, dan y adalah *output* sistem dan u adalah *input* sistem. Sehingga *state space* sistem dapat di rumuskan sebagai berikut.

$$\begin{aligned}
 \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-b(I+ml^2)}{I(M+m)+Mml^2} & \frac{m^2l^2g}{I(M+m)+Mml^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{mlb}{I(M+m)+Mml^2} & \frac{-(M+m)mgl}{I(M+m)+Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \\
 &\begin{bmatrix} 0 \\ \frac{(I+ml^2)}{I(M+m)+Mml^2} \\ 0 \\ -\frac{ml}{I(M+m)+Mml^2} \end{bmatrix} u \\
 y &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u
 \end{aligned} \tag{25}$$

E Perancangan Linear Quadratic Regulator (LQR)

Dalam kasus ini, beberapa parameter akan dimisalkan untuk mempermudah perhitungan, yaitu:

- $M = 0.74 \text{ kg}$
- $m = 0.1 \text{ kg}$
- $b = 1 \text{ N m}$
- $g = 9.8 \text{ m/s}^2$
- $l = 0.5 \text{ m}$
- $I = 0.0083 \text{ kg m}^2$

Dengan menggunakan pemisalan tersebut, persamaan (24) dan (25) dapat dituliskan sebagai berikut

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -1.3072 & 0.9608 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1.9608 & -16.1412 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 1.3072 \\ 0 \\ -1.9608 \end{bmatrix} u \tag{27}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \tag{28}$$

Uji Keterkontrolan (Controllability)

Uji kekontrolan dari sistem ini dapat dilihat dari *rank* dan *singularity* dari matriks *controllability*, yaitu:

$$[B \ AB \ \dots \ A^{n-1}B]$$

Maka matriks *controllability* dari sistem *balancing robot* ini berdasarkan *state-space* pada persamaan (27) dan (28), adalah:

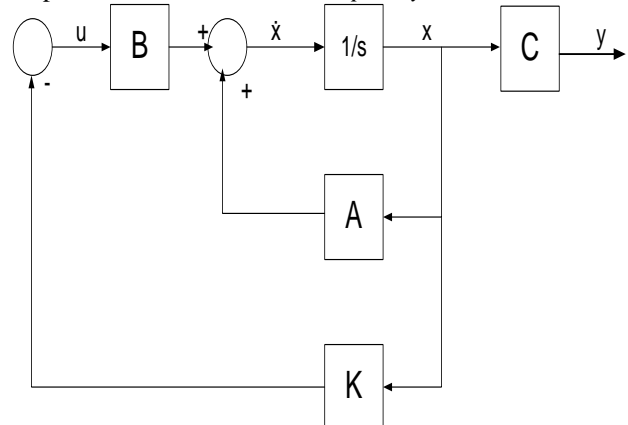
$$\begin{aligned}
 &[B \ AB \ A^2B \ A^3B] \\
 &= \begin{bmatrix} 0 & 1.3072 & -1.7087 & 0.3498 \\ 1.3072 & -1.7087 & 0.3498 & 2.0054 \\ 0 & -1.9608 & 2.5631 & 28.2989 \\ -1.9608 & 2.5631 & 28.2989 & -40.6859 \end{bmatrix}
 \end{aligned}$$

Setelah diperiksa, matriks ini memiliki *rank* 4. Sedangkan inversnya adalah:

$$\begin{aligned}
 &[B \ AB \ A^2B \ A^3B]^{-1} \\
 &= \begin{bmatrix} 1.0000 & 0.8400 & 0.0000 & 0.0500 \\ 0.8400 & 0.0680 & 0.0500 & 0.0454 \\ 0.0680 & 0.0520 & 0.0454 & 0.0347 \\ 0.0520 & -0.0000 & 0.0347 & -0.0000 \end{bmatrix}
 \end{aligned}$$

Dengan demikian matriks ini adalah matriks yang non-singular.

Karena matriks *controllability* memiliki *rank* 4, sama dengan dimensinya (matriks 4x4), dan sudah diperiksa bahwa matriks ini adalah matriks yang non-singular, maka dapat disimpulkan bahwa sistem ini *completely state controllable*.



Gambar 9. Diagram blok sistem loop tertutup

Uji Keteramatan (*Observability*)

Uji keteramatan dari sistem ini juga dapat dilihat dari *rank* dan *singularity* dari matriks *observability*, yaitu:

$$\begin{bmatrix} C \\ CA \\ \dots \\ CA^n \end{bmatrix}$$

Uji keteramatan yang pertama adalah untuk output berupa posisi *cart balancing* robot, maka dalam hal ini:

$$c = [1 \ 0 \ 0 \ 0]$$

Matriks *observability* yang diperoleh adalah:

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1.3072 & 0.9608 & 0 \\ 0 & 1.7087 & -1.2559 & 0.9608 \end{bmatrix}$$

Setelah diperiksa, matriks ini memiliki *rank* 4. Sedangkan inversnya adalah:

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -0 & -0 \\ 0 & 1.3605 & 1.0408 & 0 \\ 0 & 0 & 1.3605 & 1.0408 \end{bmatrix}$$

Karena *rank* yang diperoleh adalah 4, sama dengan dimensi dari matriks *observability* (3x3), kemudian karena inversnya ada, sebagai indikasi bahwa matriks *observability* ini adalah matriks yang *non singular*, maka dapat disimpulkan bahwa sistem dengan output posisi *cart* ini *completely observable*.

Uji keteramatan selanjutnya adalah untuk output berupa sudut pendulum, maka dalam hal ini:

$$c = [0 \ 0 \ 1 \ 0]$$

Matriks *onservability* yang diperoleh adalah:

$$\begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1.3072 & 0.9608 & 0 \\ 0 & 1.7087 & -1.2559 & 0.9608 \end{bmatrix}$$

Setelah diperiksa, invers matriks *observability* dari sudut pendulum tidak ada karena determinan dari matriks ini adalah 0 yang menandakan matriks ini adalah matriks *singular*. Maka dapat disimpulkan bahwa sistem dengan output sudut pendulum ini *not completely observable*.

Penentuan Matriks Bobot

Matriks bobot adalah matriks *Q* dan *R*. Pemilihan matriks *Q* dan *R* dapat dilakukan dengan cara coba-coba (*trial and error*). Dengan syarat, Matriks *Q* adalah matriks simetri, semidefinit positif dan *real* ($Q \geq 0$). Matriks *Q* merupakan matriks berordo 4x4 yang ditulis sebagai persamaan (29).

$$Q = \begin{bmatrix} q & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & q & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{29}$$

Pada *Q* di dapat dari $Q = C^T C$, sehingga nilai yang akan diganti-ganti pada matriks *Q* terdapat pada matriks $Q(1,1)$ dan $Q(3,3)$.

Sedangkan matriks *R* adalah matriks simetri, definit positif dan *real* ($R > 0$). Matriks *R* merupakan matriks berordo 1x1 yang ditulis sebagai persamaan (30).

$$R = [r] \tag{30}$$

Untuk menghitung besarnya nilai penguatan (*gain*) optimal *K* digunakan bantuan program MATLAB. Setelah mendapatkan penguatan optimal *K*, maka diperoleh persamaan keadaan untuk sistem *loop* tertutup sebagai berikut:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & \frac{-b(I+ml^2)}{I(M+m)+Mml^2} & \frac{m^2 l^2 g}{I(M+m)+Mml^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{mlb}{I(M+m)+Mml^2} & \frac{-(M+m)mgl}{I(M+m)+Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} - \begin{bmatrix} 0 \\ \frac{(I+ml^2)}{I(M+m)+Mml^2} \\ 0 \\ -\frac{ml}{I(M+m)+Mml^2} \end{bmatrix} \begin{bmatrix} K_{(1,1)} & K_{(1,2)} & K_{(1,3)} & K_{(1,4)} \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} \tag{31}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \tag{32}$$

Diagram blok yang mewakili persamaan (31) dan (32) ditunjukkan dalam gambar 9.

IV. ANALISA DAN PEMBAHASAN

A Hasil Perhitungan Nilai Umpan Balik LQR

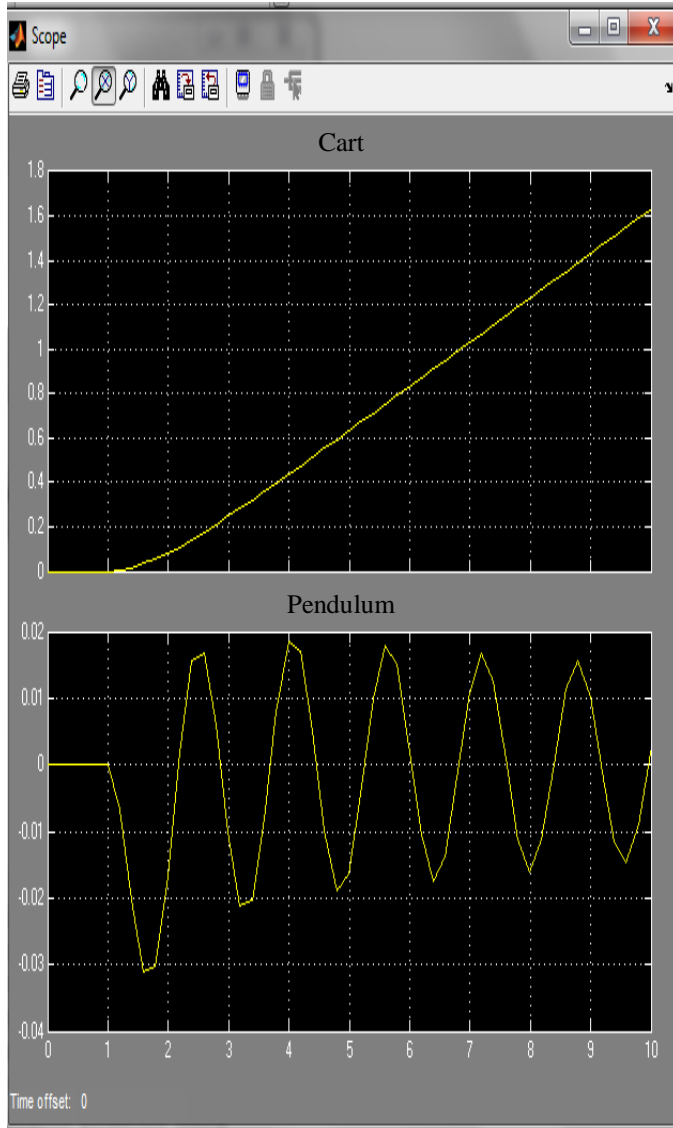
Tujuan dari optimasi dengan menggunakan LQR adalah mendapatkan nilai balik *K* optimal dengan memenuhi kriteria waktu turun (*settling time*) untuk *cart* dan pendulum kurang dari 5 detik dan waktu naik (*rise time*) untuk *cart* kurang dari 2 detik.

Perhitungan ini dilakukan dengan jalan memasukkan persamaan *Riccati* yang telah diturunkan pada bab sebelumnya. Sedangkan matriks pembobotan *Q* dan *R* ditentukan secara sembarang. Dari persamaan *Riccati* tersebut, akan dapat diketahui matriks *P*.

Matriks *P* adalah matriks solusi dari persamaan *Riccati*. Jika nilai matriks *P* telah diketahui, kemudian disubstitusikan ke persamaan $K = R^{-1} B^T P$. Sehingga dapat diketahui nilai matriks umpan balik optimal *K*. Bobot-bobot nilai *Q* dan *R* yang digunakan terdapat pada tabel I.

TABEL I HASIL PERHITUNGAN NILAI UMPAN BALIK LQR

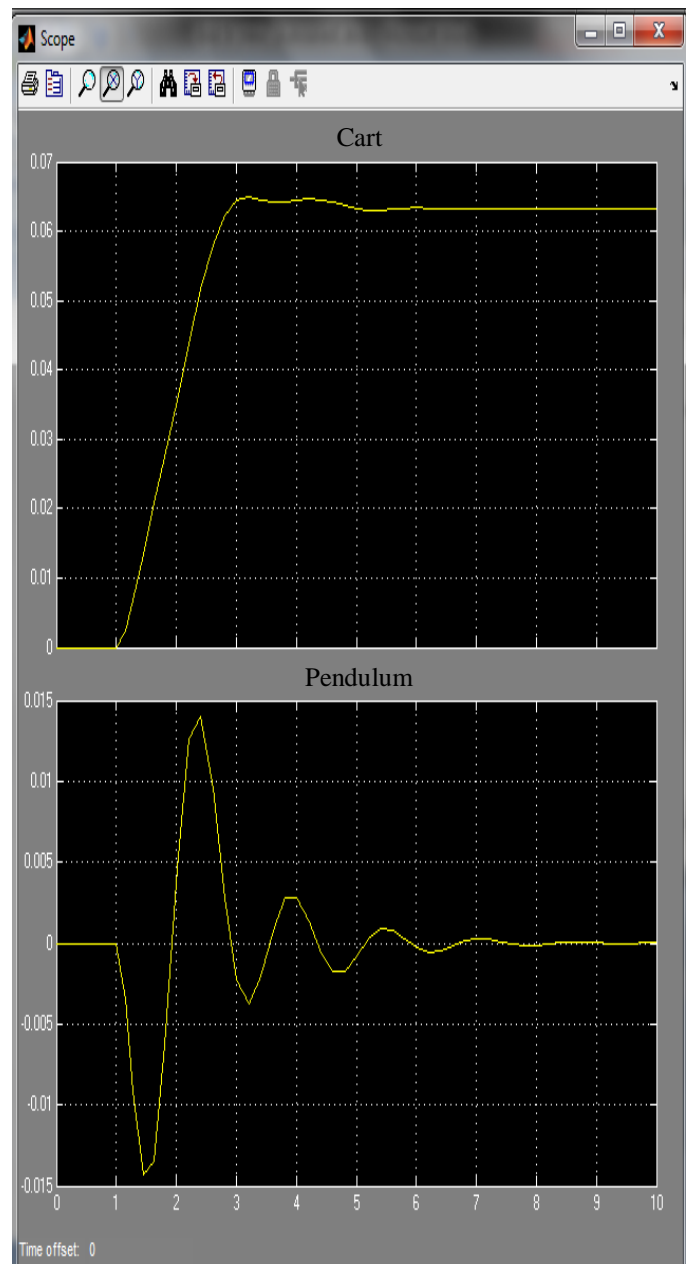
Q(1,1)	Q(3,3)	R	K			
0.1	0.1	10	0.1000	0.0808	-0.0068	-0.0001
0.5	0.5	10	0.2236	0.1735	-0.0321	-0.0109
1	1	10	0.3162	0.2390	-0.0602	-0.0231
5	5	10	0.7071	0.4905	-0.2322	-0.0844
10	10	10	1.0000	0.6619	-0.4009	-0.1304
25	25	10	1.5811	0.9792	-0.8084	-0.2116
50	50	10	2.2361	1.3182	-1.3529	-0.2835
75	75	10	2.7386	1.5719	-1.8136	-0.3242
100	100	10	3.1623	1.7833	-2.2224	-0.3494



Gambar 10. Sistem Untuk *Open Loop*

B Simulasi Balancing Robot dengan Simulink MATLAB

Untuk melihat simulasi dari *balancing robot*, dilakukan dengan menggunakan simulink MATLAB. Simulasi diberi sinyal uji step dengan perpindahan posisi dari kereta sebesar 0.2 meter ke arah kanan.



Gambar 11. Sistem Untuk $Q(1,1) = 100$, $Q(3,3) = 100$ Dan $R = 10$

Simulasi Open Loop

Untuk simulasi *open loop* atau tanpa umpan-balik, hasil simulasi menunjukkan bahwa sistem tidak stabil. Hasil ini dapat dilihat pada gambar 10.

Simulasi Close Loop

Simulasi ini dilakukan pada berbagai nilai bobot Q dengan nilai R yang tetap, dengan menggunakan MATLAB. Simulasi diberi nilai $Q(1,1) = 100$, $Q(3,3) = 100$, dan $R = 10$. Sehingga diperoleh grafik respon sistem terhadap sinyal uji *step* seperti pada gambar 11.

Pada Gambar 11 dapat diamati, Sistem memiliki waktu stabil (*settling time*) untuk *cart* 3.8 detik dengan kestabilan pada jarak 0.063 dan untuk pendulum 4 detik. Waktu tunda (*delay time*) untuk *cart* 1.8 detik.

TABEL II HASIL SIMULASI SISTEM *BALANCING* ROBOT

Q(1,1)	Q(3,3)	R	Waktu Stabil cart (s)	Waktu Stabil Pendulum (s)	Waktu Naik cart (s)
0.1	0.1	10	76.2	56	76.2
0.5	0.5	10	34.7	43.4	34.7
1	1	10	22.4	34.8	22.4
5	5	10	7.9	18.2	7.9
10	10	10	15.7	11.8	5
25	25	10	5.6	7.8	3.2
50	50	10	5.2	7.4	3.6
75	75	10	3.9	4.8	2.1
100	100	10	3.8	4	1.8

V. KESIMPULAN DAN SARAN

A Kesimpulan

Berdasarkan dari hasil analisa yang terdapat pada tabel II diperoleh kesimpulan yaitu hasil optimal dengan menggunakan metode LQR memiliki respon *balancing* robot yang paling optimal dengan kriteria yang diberikan terdapat pada nilai matriks bobot $Q(1,1) = 100$, $Q(3,3) = 100$ dan $R = 1$

B Saran

Pemilihan matriks bobot Q dan R pada penelitian ini dilakukan dengan cara coba-coba (*trial and error*), dengan membutuhkan waktu yang cukup lama untuk mendapatkan hasilnya. Oleh sebab itu perlu dikembangkan metode yang lebih baik untuk mendapatkan nilai matriks Q dan R tersebut. Metode ini dapat dikembangkan dan dapat diterapkan pada sistem fisik *balancing* robot.

DAFTAR PUSTAKA

- [1] F, Gene. Feedback Control Of Dynamic Systems, Prentice Hall, California, Fourth Edition.
- [2] K, Ogata. Modern Control Engineering, Prentice Hall, New Jersey, Third Edition,.
- [3] K, Ogata. Teknik Kontrol Automatik , Penerbit Erlangga, Jakarta, 1996.
- [4] N, S, Desineni. Optimal Control Sitemns, CRC Press, New York, 2003.