TÍTULO DA TESE

Nome do Autor Sobrenome

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Civil, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Civil.

Orientadores: Nome do Primeiro Orientador Sobrenome
Nome do Segundo Orientador Sobrenome
Nome do Terceiro Orientador Sobrenome

Rio de Janeiro
Fevereiro de 2011

TÍTULO DA TESE

Nome do Autor Sobrenome

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ
COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE)
DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS
REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR
EM CIÊNCIAS EM ENGENHARIA CIVIL.

Examinada por:

_____
Prof. Nome do Primeiro Examinador Sobrenome, D.Sc.

_____
Prof. Nome do Segundo Examinador Sobrenome, Ph.D.

_____
Prof. Nome do Terceiro Examinador Sobrenome, D.Sc.

_____
Prof. Nome do Quarto Examinador Sobrenome, Ph.D.

_____
Prof. Nome do Quinto Examinador Sobrenome, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
FEVEREIRO DE 2011

A alguém cujo valor é digno
desta dedicatória.

Agradecimentos

Gostaria de agradecer a todos.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários
para a obtenção do grau de Doutor em Ciências (D.Sc.)

TÍTULO DA TESE

Nome do Autor Sobrenome

Fevereiro/2011

Orientadores: Nome do Primeiro Orientador Sobrenome
 Nome do Segundo Orientador Sobrenome
 Nome do Terceiro Orientador Sobrenome

Programa: Engenharia Civil

Apresenta-se, nesta tese, ...

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

THESIS TITLE

Nome do Autor Sobrenome

February/2011

Advisors: Nome do Primeiro Orientador Sobrenome
          Nome do Segundo Orientador Sobrenome
          Nome do Terceiro Orientador Sobrenome

Department: Civil Engineering

In this work, we present ...

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Relatório

In this thesis an observer for an active prosthetic leg will be delevoped using an observer approach designed by Khalil. The observer has an adaptive gain related to joints signal-to-noise ratios. The simulation step used in this thesis is 10Khz and a real-time controller with 1kHz sampling rate was considered in order to implement the real control signal.

In order to define a real SNR, one must consider the resolution error or the data provided on the datasheet.

2018-07-09

There 3 possible approaches for a SNR estimation: 1) Filter the raw signal in order to obtain a signal + noise

2) Create one more state, which will be regarded as disturbances

3) Empirical method

Tests with the SNR estimation algorithm showed that the Empirical method needs some improvements to be done, probably because it was designed for a different application.

To be done

- Find the encoder used in the prosthetic leg and recreate the error signal distribution

- Improve the empirical method

  - Avoid the spikes occurring during estimation

- References

  - https://engagedscholarship.csuohio.edu/cgi/viewcontent.cgi?article=1014&context=etdarchive

  - journal of neuroengineering and rehabilitation

2018-07-11

The main goal to use encoders instead of load cells for ground reaction forces estimation is to reduce weight and cost. As a drawback, a mathematical model of external forces needs to be designed. In Fakoorian, only joint angles are considered and that's a simplification for the case of a normal gait. If anything different from a flat ground appears, i.e. ramp or stair, the prosthetic user won't be able the take a step further. The mathematical model for GRF estimation is a point of improvement and could also have motor currents as inputs.

Fakoorian was the first to propose an encoder as replacement for load cells as a GRF data device.

To be done

- Describe types of prosthesis and main suppliers/developers

- Define an encoder resolution to use in the simulation with noise

-

2018-07-19

Tests with lowpass filter have been done in order to estimate the output signal SNR. Firts of all, the input SNR has to be known so one could compare with a given SNR estiamtion. A simulation that considers quantization, noise and sampled-data implementation has been developed.

In order to define design a first order low-pass filter, the signal FFT has been calculated as depicted in **??**. Based on it, it has been concluded that a cutoff frequency in 30Hz would be enough to filter and provide an acceptable signal phase-shift.

2018-08-16

Test with the first order filter will be carried on to validate it's implementation. First of all it will be designed the filter with a cutoff frequency at 30Hz then an SNR algorithm will acquire the noise signal by subtracting the raw_signal from the filtered_signal.

2018-08-17

A chapter for SNR estimation has been created. In that file the whole procedure for estimating SNR will be explained

2018-08-18 The filter has been implemented and a the data was stored as *data_-steps_filtered* The joint unit is an angle in radians.

The results with the first order lowpass filter for a 15sec simulation were: due to a phase-shift of [1.50791.47781.47451.5059] degrees, the filtered signal had a maximum joint error of [1.146.3512.039.17] degrees and the same SNR was obtained.
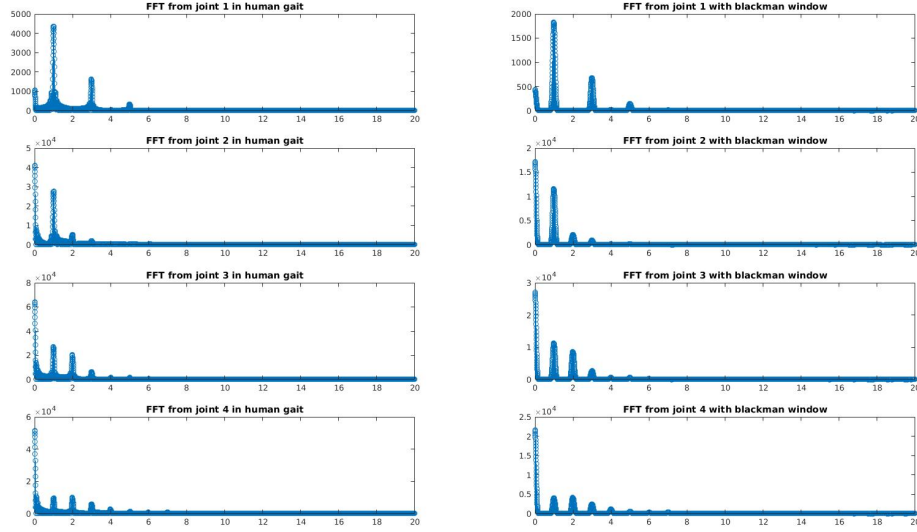
Figura 1.1: FFT example for Human gait considering a 1sec step

Another implementation would be the empirical one using the script *sinal-ruido.m*. 2 Bounderies are computed during the simulation, an upper and a lower one, in order to estimate a noise. The outputs from this script are $[SNR\,ruido\,upper\,lower\,noise]$

2018-08-20

The empirical approach was tested with some script changes. For the second joint an error of 7.3830 was obtained according to euclidian distance (norm command in Matlab) and a maximum 8.35 degrees error during a 5 seconds simulation.

2018-08-24

Considering an encoder resolution of 12 bits, the measurement precision is equal to $\frac{180}{\pi} \cdot \frac{1}{2^{12}} = 0.014$, that means my quantization error would be more or less half my precision (0.007). Using the *SNR_mod.m*, it is possible to set the noise variance to 0.007 and acquire the noisy signal equivalent to that.

Today I had a meeting with Jacoud and we set some tasks about the thesis as follows:

1. Run simulation with 4 joints tracking a reference signal (*data_steps*) with the states (from plant) and computed torque control without ground reaction forces

   (a) Create parametrical errors in matrix used for computed torque

2. Run simulation with 4 joints tracking a reference signal (*data_steps*) with estimated states (from HGO) and computed torque control without ground reaction forces

3. Implement 2 or 3 forms to estimate in open loop the SNR from signal control

   (a) Jacoud

   (b) Low-pass (design filter using previous control signals)

4. Implement a block in simulation that has error norm and SNR as inputs and a variable gain as output without ground reaction forces

It is important to say the changes regarding the project. First of all, the SNR estimation will be used at the control signal instead of joint angles. It means that my control signal will have some high frequency components that will be filtered in order to smooth it. However, there is no a priori knowledge of its behavior. Therefore some samples will be needed in order to specify its frequency components.

Another change is regarding the rate trasition. As the simulation should increase its complexity according to some steps, the implementation of different sample frequencies for a real simulation will be postponed.

The first step today will be define every unit used in the system, as torque, angles, etc.

Units Table	
Torque	N.m
Mass	Kg
Subject length	m
Joint angles	rad
Force	m/s^2

After using the HGO states, remember to use a saturation block at the control signal, otherwise it will "explode".

2018-08-26

After simulating the first task (State Feedback), one could see that:

• Initial state mismatch could lead to a transient response in the control signal

•

2018-08-27

A simulation routine using the *save* command must be implemented in order to automate the parameters value change.

2018-08-31

After some trials, the need for correct control parameters was obvius. To do that the following control project was designed.

$$M\ddot{\theta} + C\dot{\theta} + G = \tau \tag{1.1}$$

$$\tau := C\dot{\theta} + G + \bar{\tau} \tag{1.2}$$

$$\bar{\tau} := M(u) \tag{1.3}$$

$$\ddot{\theta} = u \tag{1.4}$$

$$e = \theta - \theta_d \tag{1.5}$$

$$\ddot{e} = u - \ddot{\theta}_d \tag{1.6}$$

$$u := \ddot{\theta}_d - K_p e - K_d \dot{e} - K_i \int_{t_o}^{t} e \, dt \tag{1.7}$$

$$\dddot{e} + K_d \ddot{e} + K_p \dot{e} + K_i e = 0 \tag{1.8}$$

$$s^3 + K_d s^2 + K_p s + K_i = 0 \tag{1.9}$$

$$(s^2 + 2\zeta\omega_n + \omega_n^2)(s + p) = 0 \tag{1.10}$$

$$Ki = \omega_n^2 p \tag{1.11}$$

$$K_p = \omega_n^2 + 2\zeta\omega_n p \tag{1.12}$$

$$K_d = p + 2\zeta\omega_n \tag{1.13}$$

As the system's highest frequency component is 30Hz, $\omega_n > 2\pi 30$. The damping factor will be arbitrarily chosen as $\zeta = 0.9$ and the pole should be faster than the ω_n, therefore ... $3\omega_n$.

2018-08-31

The control input signal is saturating right in the beginning of the simulation. After some tests regarding dqInit and qInit conditions, it seems that the dirty derivative may be causing a high joinv velocity reference signal and this may cause $K_d\dot{e}$ to be a high value. This conclusion was taken after changing tau from 0.01 to 0.001 which gave the controller a better performance (reduced the tracking error from 0.1rad to 0.01 in PID)

One possible solution may be differentiating the data_steps manually instead of applying durty derivative.

Now one must create the *.mat* files with simulation results and then move to another step from the thesis (add parametrical error and then estimated states feedback)

Test with parametrical error have been implemented in *Mass_ctrl Inertia_ctrl R_ctrl L_ctrl h_ctrl* and both PD and PID control could handle it. After adding and error of $1 + 0.5 * (rand() - 1/2)$ the tracking error has the same peak values although the dynamic of it has changed.

During the implementation of HGO states feedback one could see that the con-

trol signal was saturating. That happened because the HGO gain μ was 0.0001, after changing its value to 0.001 the control signal had a better behaviour and the performance was the same.

2018-09-01

After noticing that the dirty derivative has its limitations, another approach to differentiate joint signals could be by curve fitting. That can be done using a matlab command called *cftool*, in which one must enter the period and the signal, then choose the fitting curve that best suits the application. In this thesis case, the *sum os sines* fit would be the best. For some joints the curve dynamics require more than 8 sines to be accurately represented, therefore this method was disregarded for now.

2018-09-04

Today the focus was on Empirical SNR Estimation method. After some problems regarding the superior and inferior buffer, one could conclude that as the buffers are declared as global variables then every function in which they are called updates the same buffer instead of having local alterations. One way to correct that can be creating one function for each joint in order to avoid this global declaration problem.

Another issue could be the noise signal that is calculated as $(sup - inf) - sinal$ in *sinalruido.m*. Probably the best way to do this would be using the raw signal - this filtered signal from the script $(u - sinal)$.

2018-09-05

Today a simulation similar to the script *sinalruido.m* was created and is working. This simulation also implements the power calculation.

Another issue I had is regarding the *calcPower.m* script, because the control signal has high power during the first 20ms of simulation and that had changed the actual control signal power. One way to solve that could be using step period as input and then calculate the mean power with each step period. Another approach could be ... disregard the first step and then use the other measures. Either way ... the simulation in which a signal power is calculated is working properly.

One must pay attention to the buffer size otherwise the power from the raw signal and the filtered one will be way different.

2018-09-06

The SNR estimation was implemented today and the estimation error has to be implemented yet. It seems that the results are good and now a comparison between the empirically filtrated signal and the low pass filtrated signal SNR.

2018-09-10

Tests with SNR estimation have been done in order to set an average SNR for each control signal. A problem regarding the noise has arised as the noise is calculated subtracting real signal from the filtered one and the second is dependent

on window size, in other words, the raw SNR changes according to the window size as well. It becomes tricky to set this size, so a script will be run changing the window size values to relate accordingly to the SNR.

2018-09-13

After trying to run the simulation with HGO state feedback one could see a noisy control signal due to a high parametric error (25%), the solution was a reduction to 10%.

After a meeting with Jacoud, it was decided that no proper SNR will be measured. The noise measure is $u_{max} - u_{min}$ and some other parameters will com from that. The idea is to create a noise measure block where the control signal is the input and there will be 4 outputs:

- $abs(u_{max} - u_{min}) = \Delta$

- Δ filtered with $\frac{1}{s+1}$

- $\frac{\Delta}{control\_signal_{mean}}$

- $\frac{\Delta}{control\_signal_{mean}}$ filtered with $\frac{1}{s+1}$

After ploting the results, it can be seen that the filtered signals have smoothier behaviours and therefore are more likely to be used, otherwise the μ from observer will change its values in high frequency.

2018-09-19

A comparison between HGO and EKF RMSE has been taken and one could see EKF having a beter performance in joint velocity estimation, probably because of a wrong tunning in HGO parameters. In that way, an **HGO design has to be made**.

Besides that, an article to ACC regarding this SNR x Tracking error approach is being made.

2018-09-20

Test with the simulation have been conducted to create an HGO observer with varying gain. However the older implementation couldn't handle that. Instead of using a state space block the whole block diagram has been implemented, the dynamics is the same but allowed a variable HGO gain implementation.

Another change is regarding the dirty derivative. A model reference has been designed to have a filtered gait data and its derivatives, performing a better approximation than the dirty derivative approach.

In order to define the minimum and maximum HGO gain, trials have been made in which the conclusion is ... the gain should be in between $0.5e - 3$ and $1.4e - 3$.

2018-09-20

Experimentally there are peaks in the first 15ms of simulation due to the HGO Observer. To reduce these peek effects saturation blocks have been added at the SNR estimator.

Another modification is regarding the delta, a.k.a noise, and the SNR inclusion. The delta signal is composed of 4 different ways to "estimate"SNR for one joint, but the main goal is to separate 4 SNR estimation methods, each one composed of the 4 joints SNRs. Therefore a system os MUX and DEMUX has been created in a way that SNRn block is now made of each joint SNR method-n.

After some other HGO gain tests it has been seen that the gain range is from $0.4e-3$ to $1.9e-3$.

2018-09-22

Now there is a variable gain with a gain in its equation to adjust the fraction value. It can be seen that changing the noise value will also make the variable gain converge to a different value. The higher the noise, the bigger the gain has to be.

2018-09-25

Some scripts had to be modified to use for the paper. First of all, the variable gain used by the observer is in the form

$$\mu_{var} = \frac{\alpha|noise|}{1 + \beta|error_{track}|} \tag{1.14}$$

α and β were chosen so that the gain had a smooth behavior while converging to a value between $0.4e-3$ and $1.9e-3$.

Another change is regarding the reference signal, in which the position used to be the data retrieved from human kinetics and its time derivatives were obtained through "dirty derivative" approach.

The approach currently applied uses a second-order filter with transfer function

$$H(s) = \frac{w_n^2}{s^2 + 2\zeta\omega_n + \omega_n^2} \tag{1.15}$$

so that the reference signal is actually a filtered version of human gait angles. Through this approach, time-varying derivatives ($\dot{y}_d$ and $\ddot{y}_d$) from the filtered signal are easily acquired and with no error or delay.

A plot script has also been written.

2018-09-26

As the main goal is to estimate the ground force reactions and use that information in control law, tests with the real state feedback system have been conducted. A good set of parameters are

$$s_z = 2.165; \tag{1.16}$$

$$k_b = 5000; \tag{1.17}$$

$$beta = 10; \tag{1.18}$$

$$\tag{1.19}$$

and

$$BodyContact = [44]; \tag{1.20}$$

$$contact_point_h = [0.1, 0, 0.1]'; \tag{1.21}$$

$$contact_point_t = [-0.3, 0, 0.1]'; \tag{1.22}$$

$$BodyContactPositions = [contact_point_h, contact_point_t]; \tag{1.23}$$

This way a GRF can be seen similar to one depicted in `http://www.prs-research.org/files/TFA.pdf`

2018-09-27

Interesting URLs for prosthesis information

1. `http://www.cpousa.com/prosthetics/lower-extremity/`

2. `https://biomech.media.mit.edu/portfolio_page/powered-ankle-foot-prostheses/`

3. `https://www.scheckandsiress.com/patient-information/care-and-use-of-your-device/prosthetic-knee-options/`

4. `http://www.betterlimbs.com/_uls/resources/downloads/Transfemoral_Manual-2.pdf`

5. `http://www.prs-research.org/files/TFA.pdf`

2018-09-28

Write "Current Trends in Prosthesis and Summary" with textbooks, handbooks, and trending.

2018-11-27

Some tests with HGO were made in order to create GRF estimations. It seems that acos function are not working properly or only the first joint angle is given to the acos function.

1. Test cos and acos function

2.

2018-12-10

In order to compute the GFR HGO, the first and second derivatives from GFR are required.

$$z_h = q_1 + l_2 cos(q_2) + l_3 cos(q_2 + q_3) + aH cos(q_2 + q_3 + q_4 - acos(\frac{ah}{aH})) \qquad (1.24)$$

$$\dot{z}_h = \dot{q}_1 + \dot{q}_2(-l_2 sin(q_2)) + (\dot{q}_2 + \dot{q}_3)(-l_3 sin(q_2 + q_3)) + (\dot{q}_2 + \dot{q}_3 + \dot{q}_4)(-aH sin(q_2 + q_3 + q_4 - acos(\frac{a}{a_1}$$

$$(1.25)$$

$$z_t = q_1 + l_2 cos(q_2) + l_3 cos(q_2 + q_3) + aT cos(q_2 + q_3 + q_4 + acos(\frac{ah}{aT})) \qquad (1.26)$$

$$\dot{z}_h = \dot{q}_1 + \dot{q}_2(-l_2 sin(q_2)) + (\dot{q}_2 + \dot{q}_3)(-l_3 sin(q_2 + q_3)) + (\dot{q}_2 + \dot{q}_3 + \dot{q}_4)(+aT sin(q_2 + q_3 + q_4 - acos(\frac{ah}{aT}$$

$$(1.27)$$

Capítulo 2

SNR

Signal-to-noise ratio (SNR) is defined as the relation between a signals power (meaninful information) and its noise power (unwanted information) and its value is normally given in decibel unit.

A continuous-time signals power can be calculated as:

$$P_{signal} = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} \left|x(t)^2\right|^2 dt \tag{2.1}$$

Or as:

$$P_{signal} = \lim_{N\to\infty} \left(\frac{1}{2N+1} \sum_{n=-N}^{N} |x(n)|^2 dt \right) \tag{2.2}$$

for discrete-time signals

Its SNR is:

$$SNR = \frac{P_s ignal}{P_n oise} \tag{2.3}$$

From equation (**??**) it is possible to know that a higher SNR means a more meaninful signal. To convert this ratio do decibel the following equation is used:

$$SNR_{dB} = 10 log_{10} \left(\frac{P_{signal}}{P_{noise}} \right) \tag{2.4}$$

In the present work there are three ways to estimate the SNR. The first approach is to denoise the raw signal with a filter and then use this denoised signal to obtain the noise one. The main difficulty is to design the proper analog filter, because the denoised signal ideally should have the same magnitude and no phase shift on any signal frequency, however that would require some sort of special filter. To simplify that, a first order filter will be used and its cutoff frequency was choosen after defining the frequency components of a gait as depicted in **??**.
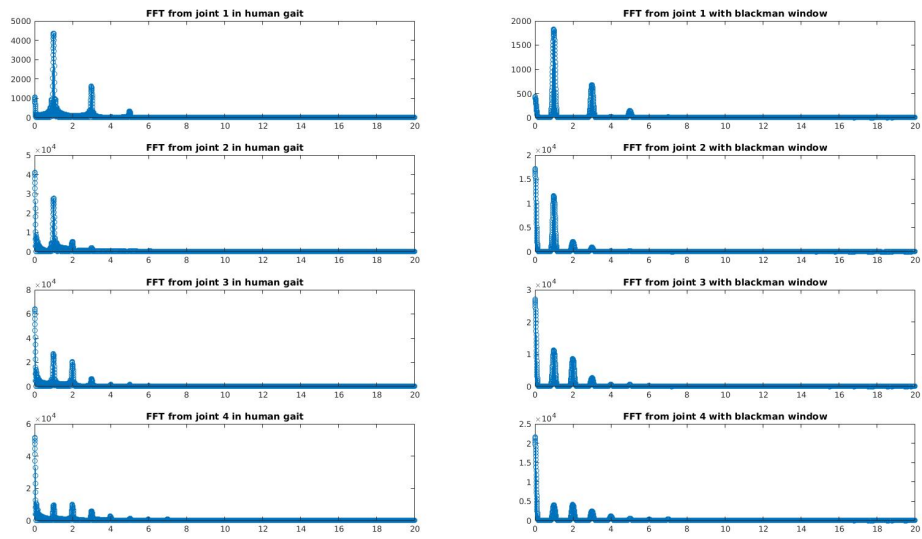
Figura 2.1: FFT example for Human gait considering a 1 sec step

Capítulo 3

Introduction

In order to better explain the control purpose applied in the simulation, a generalized control framework for active lower limb P/Os is explained (tucker2015)

High level control Mid level control Low level control

Capítulo 4

Revisão Bibliográfica

Para ilustrar a completa adesão ao estilo de citações e listagem de referências bibliográficas, a Tabela **??** apresenta citações de alguns dos trabalhos contidos na norma fornecida pela CPGP da COPPE, utilizando o estilo numérico.

Tabela 4.1: Exemplos de citações utilizando o comando padrão \cite do LaTeX e o comando \citet, fornecido pelo pacote natbib.

Tipo da Publicação	\cite	\citet
Livro	[?]	?]
Artigo	[?]	?]
Relatório	[?]	?]
Relatório	[?]	?]
Anais de Congresso	[?]	?]
Séries	[?]	?]
Em Livro	[?]	?]
Dissertação de mestrado	[?]	?]
Tese de doutorado	[?]	?]

Prosthetics have been

Capítulo 5

Método Proposto

Capítulo 6

Resultados e Discussões

Capítulo 7

Conclusões

Apêndice A

Algumas Demonstrações