



**RHYTHMIC CONTROL FOR FUNCTIONAL ELECTRICAL
STIMULATION APPLICATIONS USING DETAILED
MUSCULOSKELETAL MODELS**

FELIPE MOREIRA RAMOS

**DISSERTAÇÃO DE MESTRADO EM ENGENHARIA DE SISTEMAS
ELETRÔNICOS E DE AUTOMAÇÃO
DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

**FACULDADE DE TECNOLOGIA
UNIVERSIDADE DE BRASÍLIA**

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

**RHYTHMIC CONTROL FOR FUNCTIONAL ELECTRICAL
STIMULATION APPLICATIONS USING DETAILED
MUSCULOSKELETAL MODELS**

**CONTROLE RÍTMICO PARA APLICAÇÕES DE ESTIMULAÇÃO
ELÉTRICA FUNCIONAL USANDO MODELOS
MUSCULOESQUELÉTICOS DETALHADOS**

FELIPE MOREIRA RAMOS

ORIENTADOR: ANTÔNIO PADILHA LANARI BÓ, DR.

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA DE
SISTEMAS ELETRÔNICOS E DE AUTOMAÇÃO

PUBLICAÇÃO: PPGEA.DM-678/2017

BRASÍLIA/DF: NOVEMBRO - 2017

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

RHYTHMIC CONTROL FOR FUNCTIONAL ELECTRICAL
STIMULATION APPLICATIONS USING DETAILED
MUSCULOSKELETAL MODELS

FELIPE MOREIRA RAMOS

DISSERTAÇÃO DE MESTRADO SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:



ANTONIO PADILHA LANARI BÓ, Dr., ENE/UNB
(ORIENTADOR)



ALEXANDRE RICARDO SOARES ROMARIZ Dr., ENE/UNB
(EXAMINADOR INTERNO)



LUCIANO LUPORINI MENEGALDO, Dr., UFRJ
(EXAMINADOR EXTERNO)

Brasília, 17 de novembro de 2017.

FICHA CATALOGRÁFICA

RAMOS, FELIPE MOREIRA,

Rhythmic control for functional electrical stimulation applications using detailed musculoskeletal models [Distrito Federal] 2017.

xvi+109 p., 210 x 297 mm (ENE/FT/UnB, Mestre, Engenharia de Sistemas Eletrônicos e de Automação, 2017).

Dissertação de Mestrado – Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. Musculoskeletal Model

2. Functional Electrical Stimulation

3. Rhythmic Control

4. Coupled Oscillators

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

RAMOS, F. M. (2017). Rhythmic control for functional electrical stimulation applications using detailed musculoskeletal models, Dissertação de Mestrado, Publicação PPGEA.DM-678/2017, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, xvi+109.

CESSÃO DE DIREITOS

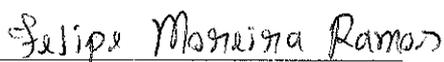
AUTOR: Felipe Moreira Ramos

TÍTULO: Rhythmic control for functional electrical stimulation applications using detailed musculoskeletal models.

GRAU: Mestre

ANO: 2017

É concedida à Universidade de Brasília permissão para reproduzir cópias desta dissertação de mestrado e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. O autor reserva outros direitos de publicação e nenhuma parte dessa dissertação de mestrado pode ser reproduzida sem autorização por escrito do autor.



Felipe Moreira Ramos

Departamento de Eng. Elétrica (ENE) - FT

Universidade de Brasília (UnB)

Campus Darcy Ribeiro

CEP 70919-970 - Brasília - DF - Brasil

For you.

ACKNOWLEDGMENTS

For my supervisor, who was always patient with me.

RESUMO

CONTROLE RÍTMICO PARA APLICAÇÕES DE ESTIMULAÇÃO ELÉTRICA FUNCIONAL USANDO MODELOS MUSCULOESQUELÉTICOS DETALHADOS

Autor: Felipe Moreira Ramos

Orientador: Antônio Padilha Lanari Bó, Dr., ENE/UnB

Programa de Pós-Graduação em Engenharia de Sistemas Eletrônicos e de Automação

Brasília, 17 de novembro de 2017

1 INTRODUÇÃO

Devido à falta de atividades apropriadas, pessoas com lesão na medula espinhal (SCI do termo em inglês) apresentam redução da massa muscular e diminuição da densidade óssea Stein (1999). Além disso, pessoas com SCI podem apresentar cicatrização lenta devido à baixa circulação de sangue, assim como pouco controle da temperatura corporal e outras funções autônomas. A vida sedentária e o enfraquecimento do corpo fizeram o número de mortes de pessoas com SCI aumentar, especialmente devido a doenças cardiovasculares Washburn and Figoni (1999).

Atividades baseadas em estimulação elétrica funcional (FES do termo em inglês) estão ajudando a reduzir os efeitos mencionados anteriormente e a melhorar parâmetros do sistema cardiovascular, como frequência cardíaca e pressão arterial Deley et al. (2014). As aplicações comuns de FES, que são focadas no movimento dos membros inferiores, são verticalização, marcha e ciclismo, mas existem outras aplicações como o remo. Pessoas com SCI podem alcançar uma melhor qualidade de vida ao melhorarem suas condições fisiológicas, incluindo a redução de doenças relacionadas.

Uma breve comparação entre as aplicações de FES e a relevância para SCI leva a conclusão de que verticalização e marcha assistidas por FES geralmente necessitam de um esforço excessivo para executar a ação desejada Kralj and Bajd (1989). Além disso, há a necessidade de um suporte para o corpo para prevenir quedas e reduzir os esforços nas pernas. Por outro lado, embora ciclismo com FES evoque menos força quando comparado com verticalização e marcha Fornusek et al. (2013), ele é mais simples e mais seguro. Além disso, ciclismo com FES também gera outros benefícios para os deficientes físicos Berkelmans (2008). Remo assistido por FES vai além e combina a estimulação dos músculos paralizados das pernas com exercício voluntário de corpo superior, o que aumenta os benefícios cardiorespiratórios quando comparado com aplicações apenas com FES Ektas et al. (2014). Uma característica adicional do ciclismo e do remo é a periodicidade do movimento. Isso permite testar estratégias específicas para esse tipo de movimento.

Entretanto, testes com um triciclo e um ergômetro para remo são demorados e precisam de muitos preparativos. Geralmente, uma sessão de ciclismo ou remo com FES que inclui de 30 a 60 minutos de exercício envolveria em torno de 30 minutos adicionais de inicialização e conclusão. O tempo disponível para experimentos pode ser menor devido à músculos fracos, espasmos e à necessidade de descansar por causa da fadiga induzida pela estimulação Hunt et al. (2002); Berkelmans (2008); Fornusek et al. (2013).

Simulações computacionais têm sido usadas para testar uma grande variedade de estratégias e parâmetros em um curto período de tempo. Os modelos variam de simples modelos de barras 2D Chen et al. (1997); Gföhler and Lugner (2004); Kim et al. (2008) até modelos musculoesqueléticos 3D Abdulla and Tokhi (2014); Li et al. (2010); Yahaya et al. (2014); Hussain et al. (2008).

Simulações também são úteis para comparar diferentes estratégias para encontrar qual é melhor e testar

diferentes configurações do ambiente. Mudar a postura, a carga e a sensibilidade dos sensores se torna possível com a adição de apenas algumas linhas de código.

Como as condições do ambiente em uma simulação computacional são controladas, pode-se utilizar algoritmos de otimização para encontrar os parâmetros dos controladores para FES. Alguns estudos anteriores utilizaram algoritmo genético (GA do termo em inglês) para melhorar a performance Abdulla and Tokhi (2014); Kim et al. (2008); Yahaya et al. (2014); Hussain et al. (2010). Yahaya et al. (2014) também usou otimização de enxame de partículas.

No caso dos estudos com modelos 3D, os projetos conhecidos usam Matlab ou Visual Nastran (Vn4D) para desenvolver controladores para FES. Esses programas possuem ferramentas para análise e visualização de dados, o que facilita a comparação de resultados. Entretanto, cada projeto usa um modelo musculoesquelético próprio. Assim, a comparação de resultados entre eles é limitada.

OpenSim é um *software* de código aberto para o desenvolvimento de modelos musculoesqueléticos e simulações de dinâmica Delp et al. (2007). Ele possui músculos, juntas e tipos de contatos predefinidos. Isso facilita o desenvolvimento de novas aplicações e comparação de resultados. Até onde sabemos, não há trabalho que utiliza OpenSim para investigar ciclismo e remo assistidos por FES.

O objetivo principal deste trabalho consiste em desenvolver e avaliar em simulações novos controladores rítmicos para exercícios assistidos por FES, como ciclismo e remo. Os objetivos específicos estão definidos a seguir:

- Desenvolvimento de *framework* para simulação de aplicações com FES utilizando modelos musculoesqueléticos;
- Usar métodos de otimização para sintonizar os parâmetros dos controladores;
- Concepção de controladores que exploram as propriedades dos movimentos rítmicos, possibilitando a utilização em outros exercícios rítmicos;
- Propor método para comparar a performance de diferentes exercícios assistidos por FES;
- Comparar a performance de diferentes controladores em diferentes modelos e diferentes configurações de ambiente.

2 MÉTODO

2.1 Modelos

Modelos musculoesqueléticos disponíveis pelo OpenSim foram modificados para simular aplicações com FES. Foram adicionados novos objetos (objetos sólidos), juntas, músculos e forças. Como o objetivo era recriar as aplicações reais, os modelos precisavam conter os equipamentos de cada exercício ou pelo menos parte deles. O *software* Blender foi utilizado para criar os novos objetos com base nas medidas dos equipamentos reais. Esses novos objetos foram adicionados aos modelos utilizando o pacote do OpenSim para Matlab¹ e foram conectados ao resto do modelo através de juntas e restrições. Os modelos musculoesqueléticos para ciclismo e remo assistidos por FES são mostrados na Figura 1.

Para simular o movimento do ciclismo, foram adicionados cinco objetos ao modelo: dois suportes para os pés, dois pedais e uma pedivela. A pedivela está fixada no *ground* e rotaciona apenas no plano sagital. Os

¹<http://simtk-confluence.stanford.edu:8080/display/OpenSim/Scripting+with+Matlab>

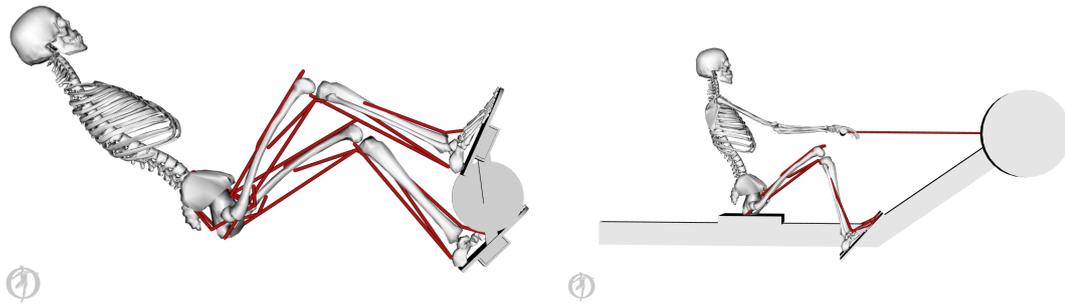


Figure 1: Modelos musculoesqueléticos para ciclismo (esquerda) e remo (direita) assistidos por FES.

pedais estão anexados no final dos braços da pedivela e rotacionam ao longo do eixo perpendicular aos braços. Os suportes são utilizados para fixar os pés nos pedais.

Para simular o movimento do remo, foram adicionados três objetos ao modelo: a estrutura base do ergômetro, um assento e uma manopla. A base está fixada no *ground*. O assento está anexado acima da base e translada ao longo do plano sagital. A manopla está anexada ao final da base e também translada ao longo do plano sagital. A manopla e a base estão conectadas por um músculo *Thelen2003Muscle*² para simular a corrente do equipamento.

A pose dos modelos também tem que assemelhar-se a uma pessoa usando o equipamento. A parte superior do corpo foi fixada para simular a posição de uma pessoa sentada. Além disso, alguns graus de liberdade foram fixados para representar a geometria dos equipamentos.

2.2 Framework para simulação de controle em malha fechada

O *framework* básico para as estratégias de controle está ilustrado na Figura 2. O bloco *Musculoskeletal Dynamics* possui como entrada um vetor (U) com a ativação de todos os músculos utilizados na estratégia.

Uma das características que faz movimentos rítmicos um problema complexo é a diferente atuação dos músculos. Cada músculo tem o seu perfil de ativação, mas eles contraem de forma coordenada para executar o movimento como um todo. Então, segmentar o movimento rítmico em estados e escolher quais músculos serão ativados pode ser uma estratégia melhor que controlar cada músculo separadamente. Por isso, um controlador de estados finitos (FSC do termo em inglês) foi inserido ao controle de alto nível para escolher quais músculos serão ativados pelo controle de baixo nível. FSC é particularmente importante para remo com FES porque o controlador tem que aplicar uma estimulação constante para manter os joelhos estendidos enquanto o participante puxa a manopla Davoodi et al. (2002). A capacidade de escolher os músculos de cada estado do movimento inclusive torna possível o uso do FSC com uma ativação constante para realizar um controle com padrão *on-off*.

Os modelos musculoesqueléticos são bastante simples quando comparados a uma pessoa. Por exemplo, eles não consideram a impedância dos tecidos e o recrutamento não intencional de músculos devido à difusão do estímulo elétrico Popović and Sinkjær (2012). Além disso, os modelos possuem menos graus de liberdade, menos músculos e o OpenSim possui uma física limitada. Por exemplo, não há atrito entre os objetos conectados por uma junta e o sistema não possui sensores imperfeitos. Assim, algumas propriedades foram adicionadas para simular condições reais. Entre elas, foram implementados ruído para simular sensores imperfeitos e um modelo de fadiga muscular. Para ciclismo com FES, foi implementado uma carga na pedivela para aumentar o esforço necessário para realizar o movimento.

²https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1Thelen2003Muscle.html

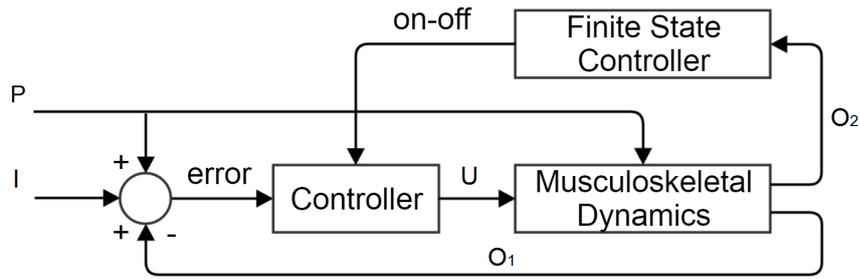


Figure 2: *Framework* básico. O bloco *Musculoskeletal Dynamics* representa a simulação do modelo do OpenSim utilizando o pacote para Matlab. O bloco *Controller* é o controle de baixo nível que muda a amplitude da ativação (U) de acordo com o erro entre a entrada (I) e a saída (O_1). O bloco *Finite State Controller* escolhe quais músculos serão ativados pelo controle de baixo nível utilizando a saída (O_2) para encontrar o estado atual do movimento. O vetor de entrada P modifica o ambiente, adicionando ruído, carga e fadiga.

Os parâmetros de alguns controladores foram sintonizados utilizando algoritmo genético. Para evitar uma solução muito especializada, o algoritmo genético avaliou apenas simulações com entrada degrau e sem modificações das propriedades da simulação. O objetivo era analisar o quanto cada controlador conseguiria se adaptar às simulações com modificações utilizando apenas os parâmetros encontrados com o algoritmo genético.

2.3 Taxa de trabalho do músculo e eficiência do custo energético

Hunt et al. (2012) recomenda o uso da eficiência do trabalho total e custo de estimulação como medidas de referência para ciclismo assistido por FES. O objetivo inicial era utilizar essas referências para comparar ciclismo e remo, pois ambos produzem trabalho mecânico e os movimentos têm um custo energético metabólico. Entretanto, essas medidas de referência são para testes reais. O custo energético metabólico do trabalho de um exercício aeróbico em regime permanente é estimado com base das taxas de absorção pulmonar de oxigênio e produção de dióxido de carbono.

O custo energético metabólico utilizado neste estudo foi calculado utilizando uma ferramenta do OpenSim baseada no trabalho de Umberger et al. (2003). A taxa total do gasto energético muscular do modelo proposto, que é expressa em W/kg de tecido muscular, é a soma das energias térmica e mecânica liberadas durante um movimento em regime permanente. A equação geral para a ativação máxima de um músculo é

$$\dot{E} = \dot{h}_{AM} + \dot{h}_{SL} + \dot{w},$$

onde \dot{h}_{AM} é a taxa de calor da ativação e manutenção, \dot{h}_{SL} a taxa de calor do encurtamento/alongamento e \dot{w} a taxa de trabalho mecânico específico da massa muscular.

Para comparar a performance do ciclismo e do remo assistidos por FES, a eficiência do custo energético foi definida como a média do trabalho total dividida pela média do custo energético metabólico:

$$\eta = \frac{\bar{w}}{\bar{E}} \times 100\%.$$

3 INVESTIGAÇÃO NUMÉRICA

3.1 Ciclismo assistido por FES

Um controle PI já foi implementado em um trabalho anterior de Sousa et al. (2016). Ele usa como entrada o erro entre a velocidade desejada e a velocidade da pedivela. O sinal de controle é então utilizado para ativar os músculos quadríceps e isquiotibiais. O perfil de ativação dos grupos de músculos da perna direita são definidos pelo FSC como intervalos de ângulos da pedivela. Os intervalos de ângulo dos grupos de músculos da perna esquerda são definidos como os intervalos para a perna direita defasados 180° .

Todas as simulações tiveram uma duração de 30 segundos e tempo de amostragem de 0.02 segundos. Durante os primeiros 20 segundos, a velocidade desejada era $200^\circ/s$. Nos próximos 10 segundos, a velocidade desejada era $300^\circ/s$. Algoritmo genético foi utilizado para sintonizar os parâmetros do controle PI e comparar a performance com a do controlador sintonizado manualmente no trabalho anterior. Além disso, algoritmo genético também foi utilizado para sintonizar os parâmetros de controladores que também ativavam o grupo muscular glúteo e que ativavam ao mesmo tempo o mesmo grupo muscular na perna oposta para reduzir a velocidade quando ela estava acima da velocidade desejada.

O algoritmo genético foi configurado com uma pequena população de 10 indivíduos e um máximo de 10 gerações. A probabilidade de cruzamento foi 0.8 e a propabilidade de mutação foi 0.15. A alta propabilidade de mutação aumenta o número de valores testados, o que aumenta a possibilidade de encontrar uma melhor solução. Elitismo foi utilizado para não perder uma boa solução encontrada antes de um novo cruzamento e mutação. Os parâmetros encontrados utilizando apenas simulações com entrada degrau e sem modificações das propriedades da simulação foram usados para avaliar a performance dos controladores em simulações com as seguintes modificações:

- Equação de primeira ordem para fadiga com valores altos das constante de tempo de fadiga e tempo de recuperação;
- *Feedback* com ruído branco para simular sensores imperfeitos;
- Carga na pedivela proporcional à velocidade para aumentar o esforço necessário para realizar o movimento;
- Modelo musculoesquelético escalado para 1.70 m e 1.90 m. A altura original era 1.80 m.

Além do controle PI, também foram investigados controle PID e controle com lógica fuzzy (FLC do inglês) adaptado de Abdulla et al. (2014). Além disso, um controle rítmico baseado em osciladores acoplados Strogatz (1994) foi desenvolvido. Osciladores acoplados têm a capacidade de adaptar o sinal de entrada do sistema de acordo com as carecterísticas do modelo. Eles foram adicionados na entrada do sistema, como mostrado na Figura 3. Ao todo são três osciladores que possuem as seguintes equações diferenciais:

$$\dot{\theta}_{c1} = \omega_{ref} + a_{c1} \text{sen}(\theta_{c2} - \theta_m), \quad (1)$$

$$\dot{\theta}_{c2} = \omega_{ref}, \quad (2)$$

$$\dot{\theta}_m = \omega_m + a_m \text{sen}(\theta_{c1} - \theta_m), \quad (3)$$

onde ω_{c1} , $\omega_m > 0$ rad/s, a_{c1} , $a_m > 0$ rad/s e ω_{ref} é a velocidade angular desejada. $c1$ representa um controlador com oscilador não uniforme, $c2$ um oscilador constante e m representa o modelo. A saída de $c1$ é a nova entrada do controlador PI.

A estratégia de ativar os músculos da perna oposta para reduzir a variação da velocidade acima da velocidade desejada e o uso do grupo muscular glúteo provou ser efetivo. O erro máximo e o RSME em regime

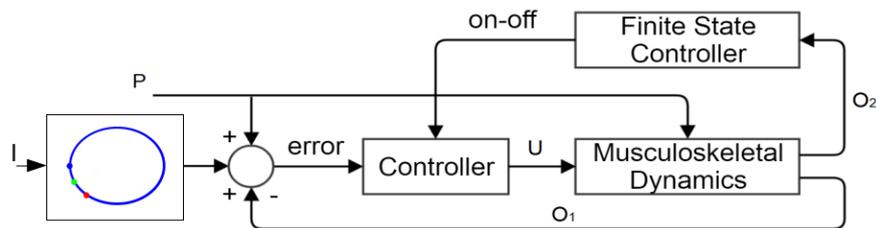


Figure 3: Diagrama do controle com osciladores acoplados para ciclismo assistido por FES. Os três pontos coloridos em uma trajetória circular representam os osciladores m , $c1$ e $c2$.

permanente são menores quando essa estratégia é comparada com outras estratégias de controle e a ativação média de um músculo apenas quando ele está ativo também foi menor. O tempo de subida da velocidade no começo da simulação foi cerca de 40-60 ms, o que não representa a resposta transitória de um músculo estimulado com FES. Em geral, OpenSim não gera fenômenos que permitem a análise da resposta transitória em sistemas FES.

O controlador PI apresentou uma performance melhor em simulações com fadiga, carga e ruído, mesmo que os parâmetros tenham sido sintonizados com simulações sem essas modificações. A adição de osciladores acoplados ao controle PI não gerou uma melhora significativa nas simulações com essas modificações, mas como mostrado na Figura 4, ao aumentar a carga na pedivela, o controlador com osciladores acoplados conseguiu manter uma velocidade média próxima à desejada enquanto o controlador PI sozinho não conseguiu.

FLC se mostrou mais estável nas simulações em que os modelos tinham diferentes escalas, como mostrado na Tabela 1. Ou seja, a performance do FLC possui valores próximos, independente da escala que foi utilizada. Assim, FLC é mais indicado para testes reais com uma grande população, onde cada um tem seu próprio peso e altura.

3.2 Remo assistido por FES

O estudo sobre remo assistido por FEs foi focado no desenvolvimento de um novo controlador rítmico. Por isso, o algoritmo genético foi utilizado apenas para sintonizar os parâmetros do novo controlador e não foram utilizadas modificações das condições da simulação.

Todas as simulações tiveram uma duração de 10 segundos e tempo de amostragem de 0.02 segundos. O FSC para remo com FES foi baseado em Davoodi et al. (2002). Os estados de um ciclo do movimento são mostrados na Figura 5. Primeiro, o modelo assume a posição *Catch*. Os músculos isquiotibiais são ativados

Table 1: RMSE dos controladores PI, PI com osciladores acoplados (PI+OSC) e FLC para uma velocidade constante de $300^0/s$. Foram considerados três escalas diferentes do modelo em simulações sem modificações do ambiente e com ruído. As simulações com carga não funcionaram corretamente para as escalas diferentes da original e por isso foram desconsideradas.

Simulação	Escala	PI	PI+OSC	FLC
Sem modificações	1.70 m	0.3272	0.4460	0.2542
	1.80 m	0.1485	0.1423	0.1795
	1.90 m	0.6152	0.5342	0.1681
Com ruído	1.70 m	0.1971	0.2049	0.2041
	1.80 m	0.1337	0.2205	0.1413
	1.90 m	0.2862	0.2561	0.1386

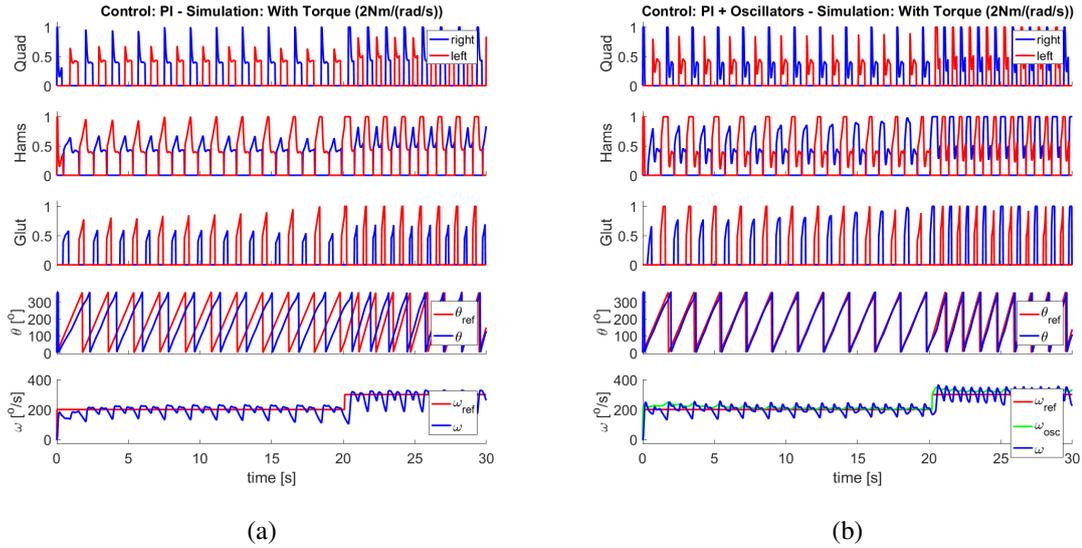


Figure 4: Resultados das simulações de ciclismo assistido por FES com *BushingForce* com coeficiente de amortecimento de 2 N·m/(rad/s) para simular carga na pedivela. Os três gráficos superiores são, de cima para baixo, o sinal enviado para os grupos musculares quadríceps, isquiotibiais e glúteo. As linhas azuis são os sinais para a perna direita e as linhas vermelhas os sinais para a perna esquerda. O quarto gráfico é o ângulo ($^{\circ}$) da pedivela. 0° corresponde à perna direita na posição superior e o sentido do movimento é horário. A linha azul é o ângulo obtido pelo OpenSim, enquanto que a linha vermelha representa o ângulo se a velocidade fosse sempre a velocidade desejada. O quinto gráfico ilustra a velocidade angular ($^{\circ}/s$) da pedivela. A linha azul é a velocidade angular obtida pelo OpenSim, a linha vermelha é a velocidade desejada e a linha verde é a saída dos osciladores acoplados. (a) mostra os resultados para o controlador PI e (b) os resultados para o controlador PI com osciladores acoplados.

para prevenir que o modelo saia da posição. O controle entra então no estado de *Drive* no qual os músculos quadríceps são ativados até a posição *Handle Pull*. Nesta posição, os quadríceps continuam ativados para que o modelo possa puxar a manopla. Depois de terminar o movimento com os braços, o controle entra no estado de *Recovery* no qual os músculos isquiotibiais são ativados e o modelo volta para a posição inicial. A posição da manopla foi utilizada para saber quando o modelo deve começar e terminar o movimento de puxar a manopla. O FSC sabe se o modelo está na posição *Handle Pull* pela posição do assento.

O FSC foi testado inicialmente com sinais constantes de ativação equivalentes a 0.25, 0.50 e 0.75, sendo que 1 corresponde à contração máxima do músculo. Os resultados obtidos foram usados como referência para comparar com os resultados do controlador desenvolvido.

Mesmo que o movimento de remo tenha um padrão rítmico, ele não pode ser classificado como um movimento puramente cíclico. Apenas os músculos das pernas são assistidos por FES, enquanto que o movimento dos braços é voluntário. O FSC espera até que o movimento de puxar a manopla termine para poder começar o estado de *Recovery*. Por isso não é possível usar um controlador de baixo nível exclusivo para movimentos rítmicos.

Alguns estudos descrevem os estados de *Drive* e *Recovery* como diferentes movimentos Davoodi and Andrews (2004); Hussain et al. (2010); Yusoff et al. (2014). O movimento de puxar a manopla também pode ser considerado como parte do estado de *Recovery* Hussain et al. (2008). Entretanto, os estados de *Drive* e *Recovery* podem ser descritos como meio ciclos de um padrão oscilatório. Além disso, o modelo tem que parar nas posições *Catch* e *Handle Pull*. Então o movimento pode ser dividido em dois movimentos discretos e uma oscilação.

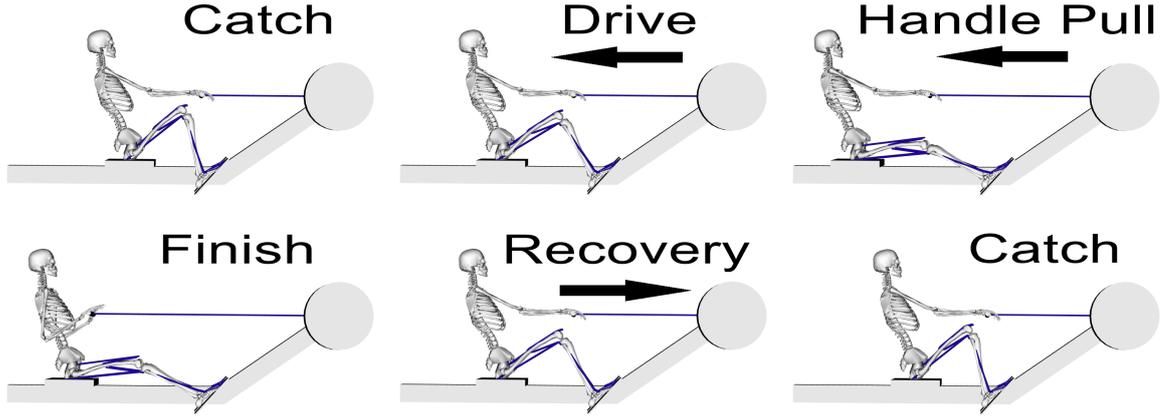


Figure 5: Estados do exercício de remo assistido por FES. O primeiro estado é o do canto superior esquerdo. Os estados seguem no sentido da direita e terminam no canto inferior direito.

Um controlador que gera movimentos discretos e rítmicos foi implementado com base no modelo de Schöner and Santos (2001). A camada capaz de gerar o sinal de saída (camada de temporização) possui três primitivas: dois atratores correspondentes ao estado inicial (posição *Catch*) e estado alvo (posição *Handle Pull*) e um oscilador Hopf Perko (2013) de amplitude 1 e frequência ω . A camada de temporização é descrita por

$$\dot{x} = -a|u_i|(x - x_i) + |u_h|(b(m - r^2)x - \omega y) - a|u_t|(x - x_t), \quad (4)$$

$$\dot{y} = -a|u_i|y + |u_h|(b(m - r^2)y + \omega x) - a|u_t|y, \quad (5)$$

onde x é o sinal de saída, y é uma variável auxiliar, $r^2 = x^2 + y^2$ e $m = 1$. a e b controlam a velocidade de convergência. i representa o atrator do estado inicial, t representa o atrator do estado alvo e h representa o oscilador Hopf. Essas primitivas são multiplicadas pelo nível de ativação $|u_k|$ ($k = i, h, t$) de três neurônios que fazem parte da camada de controle. Os neurônios são baseados no modelo de osciladores de Matsuoka (1985). A atividade neural de cada um é calculada como mostrado na Tabela 2.

Para simular a saída da camada de controle original Schöner and Santos (2001), foi utilizado $S_h = 1$, $w = 1$, $t_r = 0.1$, $t_a = 20$ e $\beta = 0.009$. A entrada S_h é sempre ativa. Ou seja, o neurônio Hopf está sempre ativo. Ele pode ser inibido quando os outros neurônios recebem um sinal de entrada diferente de 0.

No caso do remo assistido por FES, a entrada da camada de temporização é a posição do assento e a saída (x) é a ativação dos músculos. Um valor positivo de x ativa os quadríceps e um valor negativo ativa os isquiotibiais. Um diagrama do controlador com a camada de temporização e a camada de controle é ilustrada na Figura 6.

Para que o controlador consiga ajustar o nível da ativação muscular, foram implementadas três formas de *feedback*: controle da frequência (ω) do oscilador Hopf com base no erro da posição do assento, controle da amplitude (m) do oscilador Hopf com base na velocidade angular do movimento e controle da frequência (ω) do oscilador Hopf com base em osciladores acoplados. As três implementações são feitas através das seguintes equações, respectivamente:

$$\omega = (1 - q \cdot e \cdot x)\omega_{ref}, \quad (6)$$

Table 2: Atividade neural da camada de controle baseada nos osciladores de Matsuoka.

Neurônio Inicial	Neurônio Hopf	Neurônio Alvo
$t_r \dot{x}_i = -x_i - \beta v_i + S_i$	$t_r \dot{x}_h = -x_h - w u_i - w u_t - \beta v_h + S_h$	$t_r \dot{x}_t = -x_t - \beta v_t + S_t$
$u_i = \max(0, x_i)$	$u_h = \max(0, x_h)$	$u_t = \max(0, x_t)$
$t_a \dot{v}_i = -v_i + u_i$	$t_a \dot{v}_h = -v_h + u_h$	$t_a \dot{v}_t = -v_t + u_t$

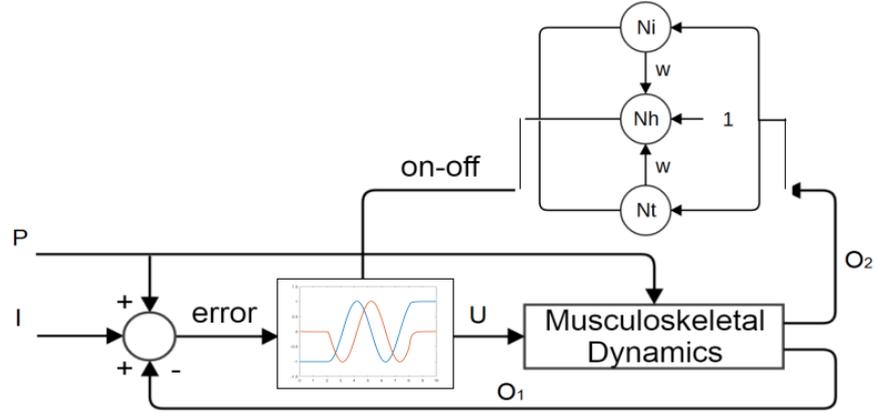


Figure 6: Diagrama do controle para remo assistido por FES baseado na dinâmica de atratores. A camada de controle é equivalente ao FSC e recebe como umas das entradas os valores de S_i e S_t . A saída é a atividade neural dos três neurônios que ativam as primitivas da camada de temporização. A camada de temporização é equivalente ao controle de baixo nível. A ativação dos músculos (U) é dada por x (linha azul).

$$m = \max(1 - (\dot{\theta}_m - \omega_{ref}) \cdot |1 - x|, 0), \quad (7)$$

$$\omega = \omega_{ref} + a_{c1} \sin(\theta_m - \text{atan2}(y, x)). \quad (8)$$

A velocidade angular $\dot{\theta}_m$ e o ângulo θ_m do movimento podem ser calculados quando o movimento é considerado como um oscilador cujo ângulo é dado pelo arco tangente da velocidade sobre a posição do assento. O controlador padrão utilizado foi implementado com o controle da frequência do oscilador Hopf com base no erro da posição do assento. Também foram implementados o controlador padrão com o controle de amplitude, um controlador com osciladores acoplados e um controlador com osciladores acoplados e controle de amplitude.

O gráfico de barras da Figura 7 mostra o número de remadas por minuto para ativação constante de 0.50 e as quatro versões do controlador baseado em dinâmica de atratores. Os controladores com osciladores acoplados apresentaram um número de remadas maior que o caso com ativação constante.

Embora a adição de controle de amplitude tenha contribuído para o aumento do número de remadas, o nível médio de ativação muscular também aumentou. Por isso, o controlador com osciladores acoplados teve a melhor performance. Além disso, a adição de osciladores acoplados conseguiu diminuir a redução abrupta da velocidade que acontecia quando o assento alcançava os limites da trajetória. A Figura 8 mostra essa redução da velocidade durante o estado de *Recovery*. Entretanto, a velocidade não diminui durante o estado de *Drive* porque o Neurônio Alvo é ativado no final da trajetória. Os osciladores acoplados só funcionam durante a oscilação.

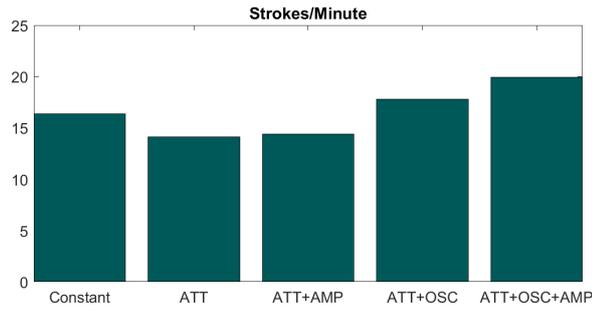


Figure 7: Gráfico de barras do número de remadas por minuto do controlador com ativação constante de 0.5 (Constant), controlador padrão baseado na dinâmica de atratores (ATT), controlador padrão com o controle de amplitude (ATT+AMP), controlador com osciladores acoplados (ATT+OSC) e controlador com osciladores acoplados e controle de amplitude (ATT+OSC+AMP).

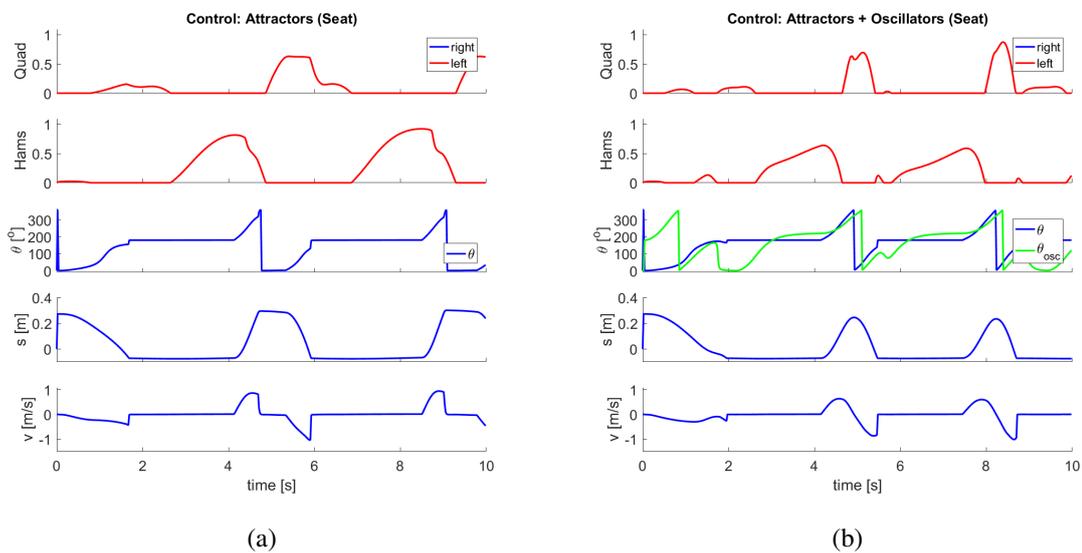


Figure 8: Resultados das simulações de remo assistido por FES com (a) o controlador padrão e (b) o controlador com osciladores acoplados. Os dois gráficos superiores são, de cima para baixo, o sinal enviado para os grupos musculares quadríceps e isquiotibiais. O terceiro gráfico é o ângulo ($^{\circ}$) do movimento. 0° é a posição *Catch* e 180° é a posição *Handle Pull*. A linha azul é o ângulo obtido através dos dados do OpenSim e a linha verde é a saída dos osciladores acoplados. O quarto gráfico é a posição (m) do assento e o quinto gráfico é a velocidade (m/s) do assento.

3.3 Eficiência do custo energético para aplicações rítmicas assistidas por FES

Não foi possível calcular a eficiência dos movimentos com base no trabalho de Hunt et al. (2012), mas a ferramenta do OpenSim baseada em Umberger et al. (2003) possibilitou comparar os exercícios de ciclismo e remo assistidos por FES. A Figura 9 ilustra o custo energético metabólico, a taxa do trabalho mecânico e a eficiência do custo energético dos controladores para ciclismo e remo que utilizaram osciladores acoplados.

Especialmente para remo com FES, foram encontrados valores altos para o custo energético. Isso aconteceu porque o modelo é expressado em W/kg de tecido muscular. Geralmente o custo energético de uma pessoa que está realizando um exercício é expressado em W/kg do peso total do corpo. Ao fazer a conversão para o peso total, os valores encontrados não passam de 20 W/kg.

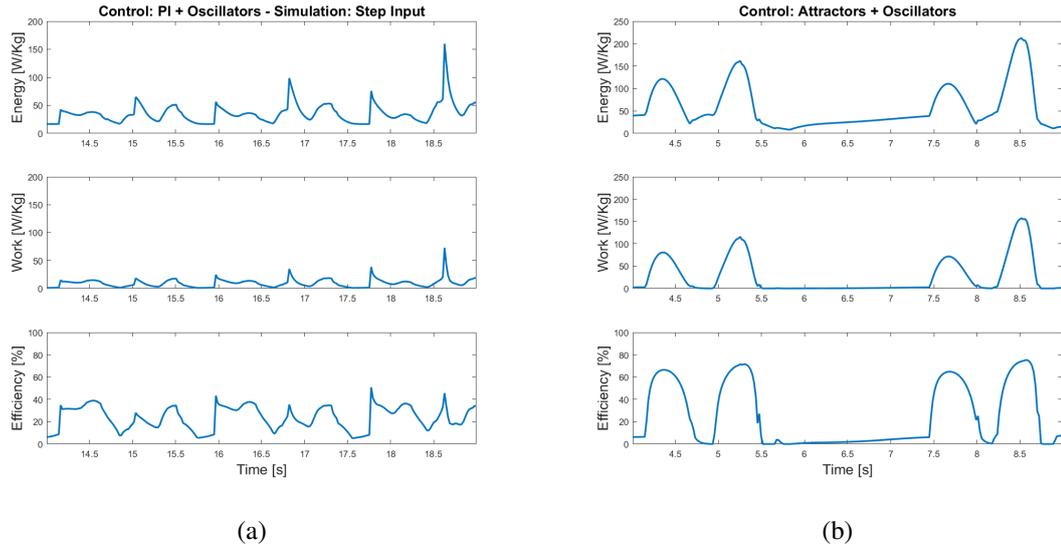


Figure 9: Custo energético metabólico, taxa do trabalho mecânico e eficiência do custo energético dos controladores para (a) ciclismo e (b) remo que utilizaram osciladores acoplados. A energia é expressada em W/kg de tecido muscular.

A eficiência média dos controladores para ciclismo com FES variou de 22% a 27%, enquanto que ela variou de 38% a 46% para remo com FES. Os maiores resultados foram com controladores que utilizaram osciladores acoplados. No caso do remo, o controlador padrão também obteve uma eficiência alta, mas se for considerado a performance geral e o número de remadas por minuto, o controlador com osciladores acoplados conseguiu melhorar a performance mantendo a mesma eficiência que o controlador padrão. A adição de controle de amplitude parece diminuir a eficiência do remo, mesmo quando o controlador também usa osciladores acoplados.

Ao considerar o custo energético metabólico e a taxa do trabalho mecânico separadamente, o exercício de remo possui altos picos de energia e inclinação acentuada durante os estados de *Drive* e *Recovery*. Por outro lado, o exercício de ciclismo produz menos calor e tem uma variação mais suave de energia. Assim, ciclismo assistido por FES é mais indicado para uma etapa inicial do condicionamento físico, pois os participantes geralmente iniciam o protocolo com músculos fracos e pouca resistência. O ciclismo permitiria realizar treinamentos mais prolongados. A inclinação da curva do trabalho para o remo indica que este exercício exige mais potência dos músculos. Ele poderia então ser utilizado após a etapa inicial do condicionamento físico para aumentar a força dos músculos e permitir a realização de exercícios que exigem mais do corpo do participante.

4 DISCUSSÃO

O objetivo deste projeto foi explorar as características de estratégias de controle rítmico para melhorar a performance dos exercícios de ciclismo e remo assistidos por FES. Para isso, foi desenvolvido um ambiente de simulação com modelos musculoesqueléticos para acelerar a prototipagem de novas estratégias de controle.

Embora os modelos musculoesqueléticos ainda sejam simples quando comparados com o corpo humano, os parâmetros encontrados utilizando algoritmo genético podem ser utilizados como um ponto de partida para encontrar os parâmetros para cada participante dos testes. Em algumas simulações houveram instabilidades que podem estar relacionadas aos cálculos realizados pelo OpenSim. Por isso é necessário um ajuste dos parâmetros do ambiente de simulação.

O controle básico das duas aplicações foi dividido em duas camadas: um controle de estados finitos de alto nível que decide os músculos que serão ativados e um controle de baixo nível que calcula o nível de ativação. No caso do ciclismo, os mesmos parâmetros foram utilizados para avaliar a capacidade adaptativa dos controladores em ambientes com fadiga, carga, ruído e para modelos com diferentes escalas.

A adição de osciladores acoplados ao controlador PI no ciclismo com FES preservou a minimização do erro e melhorou a performance nos casos em que o controlador PI sozinho não conseguiu alcançar a velocidade desejada.

A adição de osciladores acoplados também melhorou a performance do controlador baseado em dinâmica de atratores para remo assistido por FES. Esse novo controlador provou ser uma alternativa para o controle para o remo. Em vez de utilizar um controlador de estados finitos convencional, ele utiliza uma rede neural para decidir qual primitiva será ativada. A divisão do movimento em parte discreta e rítmica permitiu a utilização de estratégias de controle rítmico.

Em geral, o ciclismo assistido por FES teve uma menor ativação muscular e o modelo gastou menos energia metabólica que o remo assistido por FES. Por isso, o ciclismo é mais indicado para participantes que possuem músculos fracos e pouca resistência. Além disso, a adição de osciladores acoplados melhorou a eficiência do custo energético em ambos os exercícios. O remo pode ser usado após um condicionamento inicial com ciclismo para aumentar a força que os músculos são capazes de exercer.

Como trabalhos futuros, espera-se melhorar os parâmetros dos modelos e do ambiente de simulação, além de testar os controladores para o remo com diferentes modificações do ambiente. Adicionalmente, outros exercícios rítmicos podem ser implementados e comparados com o ciclismo e o remo. Para finalizar, espera-se usar as estratégias de controle rítmico em testes com pessoas com SCI.

ABSTRACT

RHYTHMIC CONTROL FOR FUNCTIONAL ELECTRICAL STIMULATION APPLICATIONS USING DETAILED MUSCULOSKELETAL MODELS

Author: Felipe Moreira Ramos

Advisor: Antônio Padilha Lanari Bó, Dr., ENE/UnB

Programa de Pós-Graduação em Engenharia de Sistemas Eletrônicos e de Automação

Brasília, 17th November 2017

The main objective of this work concerns the design of new controllers for rhythmic exercises assisted by functional electrical stimulation (FES), such as cycling and rowing. This work is part of a broader research effort targeting spinal cord injury rehabilitation based on principles of neuro-engineering and robotics integrated with goal-directed physical therapy. We developed a basic framework for simulation of detailed musculoskeletal models to accelerate the prototyping of new control strategies. The basic control features a higher level finite state controller and a lower level controller which calculates the activation level of the muscles. A controller with motor primitives based on attractor dynamics was developed for the rowing exercise, and coupled oscillators were added to controllers of both cycling and rowing exercises. Furthermore, we used genetic algorithms to estimate the controllers parameters, minimize the overall level of applied electrical stimulation and increase the robustness in environments with different configurations. In the case of the cycling exercise, we used the same parameters in simulations with noise, fatigue, different loads, and scales. Besides the performance evaluation, we also calculated the efficiency of the metabolic energy expenditure of the musculoskeletal models as an alternative method to compare different FES applications. Results demonstrated that the addition of coupled oscillators increased the efficiency in both exercises. Also, FES cycling seems to be more suitable for participants with motor disabilities who have weak muscles and low endurance due to its lower muscle activation and metabolic energy expenditure. FES rowing can be used later for improving the power of the muscles.

CONTENTS

1	INTRODUÇÃO	
2	MÉTODO	
2.1	MODELOS	
2.2	<i>Framework</i> PARA SIMULAÇÃO DE CONTROLE EM MALHA FECHADA	
2.3	TAXA DE TRABALHO DO MÚSCULO E EFICIÊNCIA DO CUSTO ENERGÉTICO	
3	INVESTIGAÇÃO NUMÉRICA	
3.1	CICLISMO ASSISTIDO POR FES	
3.2	REMO ASSISTIDO POR FES	
3.3	EFICIÊNCIA DO CUSTO ENERGÉTICO PARA APLICAÇÕES RÍTMICAS ASSISTIDAS POR FES	
4	DISCUSSÃO	
1	INTRODUCTION	1
1.1	CONTEXT	1
1.2	PROBLEM DEFINITION	2
1.3	OBJECTIVES	2
1.4	MANUSCRIPT ORGANIZATION	3
2	THEORETICAL FOUNDATIONS	4
2.1	MUSCULOSKELETAL SYSTEM	4
2.1.1	OPENSIM	6
2.2	FUNCTIONAL ELECTRICAL STIMULATION	8
2.3	COUPLED OSCILLATORS	9
2.3.1	DEFINITION OF OSCILLATOR	9
2.3.2	COUPLING OSCILLATORS	10
2.4	MOTOR PRIMITIVES	11
3	STATE OF THE ART	14
3.1	FES CYCLING	14
3.2	FES ROWING	15
3.3	SIMULATIONS STUDIES FOR FES CONTROL	17
4	BASIC FRAMEWORK	18
4.1	MUSCULOSKELETAL MODEL	18
4.2	FRAMEWORK FOR CLOSED-LOOP CONTROL SIMULATIONS	19
4.3	MODIFYING THE ENVIRONMENT	20
4.3.1	FATIGUE	20
4.3.2	OTHER MODIFICATIONS	21
4.4	GENETIC ALGORITHM	21
4.5	MUSCLE WORK RATE AND ENERGY COST EFFICIENCY	23
5	FES CYCLING	24
5.1	MODEL	24
5.1.1	INITIAL MODEL	24
5.1.2	ADDED OBJECTS	25
5.1.3	CYCLING POSE	25
5.1.4	LOAD	26

5.2	COUPLED OSCILLATORS FOR FES CYCLING	27
5.3	NUMERICAL INVESTIGATION	29
5.3.1	TUNING CONTROL PARAMETERS MANUALLY AND WITH GENETIC ALGORITHM	30
5.3.1.1	PREVIOUS WORK	30
5.3.1.2	METHOD	31
5.3.1.3	RESULTS AND ANALYSIS.....	32
5.3.2	USING OPPOSING MUSCLES TO REDUCE VELOCITY	36
5.3.2.1	METHOD	36
5.3.2.2	RESULTS AND ANALYSIS.....	36
5.3.3	COMPARISON WITH STATE OF THE ART	41
5.3.4	ADDITION OF GLUTEUS MUSCLE GROUP	42
5.3.4.1	METHOD	42
5.3.4.2	RESULTS AND ANALYSIS.....	42
5.3.5	STUDY ABOUT THE INFLUENCE OF LOAD.....	49
5.3.5.1	METHOD	49
5.3.5.2	RESULTS AND ANALYSIS.....	49
5.3.6	DIFFERENT SCALES	57
5.3.6.1	METHOD	57
5.3.6.2	RESULTS AND ANALYSIS.....	57
5.3.7	STUDY ABOUT THE INFLUENCE OF FATIGUE	69
5.3.7.1	METHOD	69
5.3.7.2	RESULTS AND ANALYSIS.....	69
5.4	MUSCLE WORK RATE AND ENERGY COST EFFICIENCY	77
6	FES ROWING	79
6.1	MODEL	79
6.1.1	INITIAL MODEL	79
6.1.2	ADDED OBJECTS	80
6.1.3	ROWING POSE	80
6.1.4	MOTION RESTRICTION AND KNEE REACTION FORCE	81
6.1.5	VOLUNTARY UPPER BODY MOVEMENT.....	81
6.2	NUMERICAL INVESTIGATION	82
6.2.1	AUTOMATIC CONTROL WITH CONSTANT ACTIVATION	83
6.2.2	CONTROLLER BASED ON ATTRACTOR DYNAMICS	87
6.2.2.1	METHOD	87
6.2.2.2	RESULTS AND ANALYSIS.....	89
6.2.3	AMPLITUDE CONTROL	92
6.2.3.1	METHOD	92
6.2.3.2	RESULTS AND ANALYSIS.....	93
6.2.4	ADDITION OF COUPLED OSCILLATORS	94
6.2.4.1	METHOD	94
6.2.4.2	RESULTS AND ANALYSIS.....	94
6.3	MUSCLE WORK RATE AND ENERGY COST EFFICIENCY	96
7	DISCUSSION.....	99
7.1	MODEL LIMITATIONS	99
7.2	FES CYCLING.....	100
7.3	FES ROWING.....	101
7.4	ENERGY COST EFFICIENCY OF RHYTHMIC FES APPLICATIONS	101

8 CONCLUSION	103
BIBLIOGRAPHY	105

List of Figures

- 1 Modelos musculoesqueléticos para ciclismo (esquerda) e remo (direita) assistidos por FES.....
- 2 *Framework* básico. O bloco *Musculoskeletal Dynamics* representa a simulação do modelo do OpenSim utilizando o pacote para Matlab. O bloco *Controller* é o controle de baixo nível que muda a amplitude da ativação (U) de acordo com o erro entre a entrada (I) e a saída (O_1). O bloco *Finite State Controller* escolhe quais músculos serão ativados pelo controle de baixo nível utilizando a saída (O_2) para encontrar o estado atual do movimento. O vetor de entrada P modifica o ambiente, adicionando ruído, carga e fadiga.
- 3 Diagrama do controle com osciladores acoplados para ciclismo assistido por FES. Os três pontos coloridos em uma trajetória circular representam os osciladores m , $c1$ e $c2$
- 4 Resultados das simulações de ciclismo assistido por FES com *BushingForce* com coeficiente de amortecimento de $2 \text{ N}\cdot\text{m}/(\text{rad}/\text{s})$ para simular carga na pedivela. Os três gráficos superiores são, de cima para baixo, o sinal enviado para os grupos musculares quadríceps, isquiotibiais e glúteo. As linhas azuis são os sinais para a perna direita e as linhas vermelhas os sinais para a perna esquerda. O quarto gráfico é o ângulo ($^\circ$) da pedivela. 0° corresponde à perna direita na posição superior e o sentido do movimento é horário. A linha azul é o ângulo obtido pelo OpenSim, enquanto que a linha vermelha representa o ângulo se a velocidade fosse sempre a velocidade desejada. O quinto gráfico ilustra a velocidade angular ($^\circ/\text{s}$) da pedivela. A linha azul é a velocidade angular obtida pelo OpenSim, a linha vermelha é a velocidade desejada e a linha verde é a saída dos osciladores acoplados. (a) mostra os resultados para o controlador PI e (b) os resultados para o controlador PI com osciladores acoplados.
- 5 Estados do exercício de remo assistido por FES. O primeiro estado é o do canto superior esquerdo. Os estados seguem no sentido da direita e terminam no canto inferior direito.
- 6 Diagrama do controle para remo assistido por FES baseado na dinâmica de atratores. A camada de controle é equivalente ao FSC e recebe como uma das entradas os valores de S_i e S_t . A saída é a atividade neural dos três neurônios que ativam as primitivas da camada de temporização. A camada de temporização é equivalente ao controle de baixo nível. A ativação dos músculos (U) é dada por x (linha azul).
- 7 Gráfico de barras do número de remadas por minuto do controlador com ativação constante de 0.5 (Constant), controlador padrão baseado na dinâmica de atratores (ATT), controlador padrão com o controle de amplitude (ATT+AMP), controlador com osciladores acoplados (ATT+OSC) e controlador com osciladores acoplados e controle de amplitude (ATT+OSC+AMP).
- 8 Resultados das simulações de remo assistido por FES com (a) o controlador padrão e (b) o controlador com osciladores acoplados. Os dois gráficos superiores são, de cima para baixo, o sinal enviado para os grupos musculares quadríceps e isquiotibiais. O terceiro gráfico é o ângulo ($^\circ$) do movimento. 0° é a posição *Catch* e 180° é a posição *Handle Pull*. A linha azul é o ângulo obtido através dos dados do OpenSim e a linha verde é a saída dos osciladores acoplados. O quarto gráfico é a posição (m) do assento e o quinto gráfico é a velocidade (m/s) do assento.
- 9 Custo energético metabólico, taxa do trabalho mecânico e eficiência do custo energético dos controladores para (a) ciclismo e (b) remo que utilizaram osciladores acoplados. A energia é expressada em W/kg de tecido muscular.
- 2.1 Relation between the angle (θ) and frequency ($\dot{\theta}$) of a nonuniform oscillator. The θ -axis is showing the interval $[-\pi, \pi]$ since the $\dot{\theta} = f(\theta)$ is a periodic function. The function oscillates around ω with amplitude a

2.2	Relation between the angle (θ) and frequency ($\dot{\theta}$) of a nonuniform oscillator when (a) $a < \omega$, (b) $a = \omega$ and (c) $a > \omega$. There is a saddle point when $a = \omega$, and for $a > \omega$, there are an attractor point and a repulsor point. The arrows indicate the direction of θ when t increases.	10
2.3	Relation between dimensionless angle (ϕ) and frequency ($\dot{\phi}$) of a nonuniform oscillator when (a) $\mu = 0$, (b) $0 < \mu < 1$ and (c) $\mu > 1$. For $\mu < 1$, there are an attractor point and a repulsor point. The arrows indicate the direction of ϕ when t increases.	11
2.4	Outputs X (blue) and Y (red) of the time layer. (a) is an example of rhythmic motion. (b) is an example of discrete motion with b_t working as a boolean variable. (c) is an example of discrete motion using the Equation 2.30. The initial state and target state are -1 and 1 , respectively. The amplitude of the oscillation is 1	13
3.1	Relationship between crank angle e muscle groups activation. 0° is right leg upward, and direction is clockwise. Muscle groups are right quadriceps (RQ), right hamstrings (RH), right gluteus (RG), left quadriceps (LQ), left hamstrings (LH) and left gluteus (RG).....	15
3.2	Stages of an FES rowing exercise. It starts at the top left, followed by the states in the same line, and ends at the bottom right.	16
4.1	(a) Cycling and (b) rowing models designed using Blender. Dimensions are based on measurements from real equipment.	19
4.2	Basic framework. The Musculoskeletal Dynamics block represents the simulation OpenSim performs with the Matlab scripts. The Controller block is the lower level controller that changes the amplitude of activation (U) according to the error between the input (I) and (O_1). The Finite State Controller block chooses which muscles will be activated by the lower level controller according to (O_2) to find the actual state of the dynamics. The input vector P changes the environment, i.e., adding noise, load, and fatigue.	20
4.3	Control signal (top graph) and muscle activation (bottom graph) without (blue line) and with (red line) fatigue. The maximum muscle activation is equivalent to 1 . The muscle is resting when the activation is 0 . The time constants of the fatigue equation are $T_f = 30$ s and $T_r = 30$ s.	21
4.4	Angular velocity (a) without and (b) with white noise which maximum absolute value is 5° . The red line is the desired velocity, and the blue line is the actual velocity. The modification is added to the intrinsic oscillation of the system.	22
4.5	Components of the adapted tadpole trike (HP3, Brazil). On the bottom left, a weight system to change the load (da Fonseca et al., 2016).	22
5.1	The adapted tadpole trike (HP3, Brazil). We used the measures of this tricycle to develop the additional parts of the FES cycling model (Bó et al., 2017).	24
5.2	Cycling pose. The pose of the model had to be similar to the pose in the real system (see Figure 4.5). Head-trunk and pelvis were fixed to simulate a person sitting on the tricycle.	26
5.3	System diagram for FES cycling with coupled oscillators. The three colored points in a circular trajectory represent the oscillators m , $c1$, and $c2$. The output of $c1$ is the new input of the system.	28
5.4	Relation between the angle (ϕ) and frequency ($\dot{\phi}$) of the dynamics of three pairs of coupled oscillators.	28
5.6	Phase difference (ϕ) of the dynamics of three pairs of coupled oscillator through time. ($\theta_{c2} - \theta_m$) becomes 0 rad since $\omega_{c1} = \omega_{ref}$ and the angle θ_{c1} is calculated based on the phase difference ($\theta_{c2} - \theta_m$).	29
5.5	Frequency ($\dot{\phi}$) of the dynamics of three pairs of coupled oscillator through time. The frequencies of the model and the controller with nonuniform oscillator converge to the frequency of the controller with uniform oscillator.	29

5.7	Example of a simulation result. The top three graphs are the signal sent to each muscle group, where blue lines are the signals to the right leg, and red lines are the signals to the left leg. The fourth graph is the angle ($^{\circ}$) of the crank. 0° is right leg upward, and direction is clockwise. The blue line represents the angle acquired from OpenSim, while the red line represents the angle if the velocity was always the reference velocity. The fifth graph illustrates the angular velocity ($^{\circ}/s$) of the crank. The blue line is the angular velocity acquired from OpenSim, the red line is the desired velocity, and the green line is the output of the coupled oscillators.	30
5.8	Example of a bar chart of a performance measure. The upper legend has the type of environment modification of the colors, and the groups of three bars are the performance of the control strategies.....	31
5.9	System diagram for FES cycling. The colored angle intervals of the FSC associate the crank angle with the activation of the muscles.	31
5.10	FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are a PI controller (a, c, e) manually tuned and (b, d, f) tuned by GA (PI 1). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	33
5.11	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are a PI controller manually tuned (Manual) and tuned by GA (PI 1). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, and (d) is the nonzero average hamstrings activation.	34
5.12	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are a PI controller manually tuned (Manual) and tuned by GA (PI 1). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	35
5.13	FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are (a, c, e) the previous PI controller tuned by GA (PI 1) and (b, d, f) the new one (PI 2). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.....	38
5.14	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are the previous PI controller tuned by GA (PI 1) and the new one (PI 2). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, and (d) is the nonzero average hamstrings activation.	39
5.15	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are the previous PI controller tuned by GA (PI 1) and the new one (PI 2). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	40
5.16	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.....	45
5.17	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PID controller and (b, d, f) the FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	46
5.18	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PID, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.....	47

5.19	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PID, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	48
5.20	Example of torque generated by a bushing force. The torque increased proportionally to velocity.	50
5.21	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, and with a bushing force using damping coefficient values of (c, d) 1 N·m/(rad/s) and (e, f) 2 N·m/(rad/s).	51
5.22	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a bushing force using damping coefficient values of (a, b) 3 N·m/(rad/s), (c, d) 4 N·m/(rad/s) and (e, f) 5 N·m/(rad/s).	52
5.23	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a) without environment modifications, and with a bushing force using damping coefficient values of (b) 1 N·m/(rad/s) and (c) 2 N·m/(rad/s).	53
5.24	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers is FLC (Fuzzy). We performed simulations with a bushing force using damping coefficient values of (a) 3 N·m/(rad/s), (b) 4 N·m/(rad/s) and (c) 5 N·m/(rad/s).....	54
5.25	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.....	55
5.26	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	56
5.27	FES cycling simulation results for the model which is 1.70 m tall. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	59
5.28	FES cycling simulation results for the model which is 1.70 m tall. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	60
5.29	FES cycling simulation results for the model which is 1.90 m tall. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	61
5.30	FES cycling simulation results for the model which is 1.90 m tall. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.	62
5.31	Bar charts for the model which is 1.80 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.	63

5.32	Bar charts for the model which is 1.70 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.	64
5.33	Bar charts for the model which is 1.90 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.	65
5.34	Bar charts for the model which is 1.80 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.....	66
5.35	Bar charts for the model which is 1.70 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.....	67
5.36	Bar charts for the model which is 1.90 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.....	68
5.37	FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are (a, b) manually tuned, (c, d) the previous PI controller tuned by GA (PI 1) and (e, f) the new one (PI 2). We performed simulations with fatigue and (a, c, e) $T_r = 30$ s and (b, d, f) $T_r = 60$ s.	71
5.38	FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. The controllers are (a, b) the PI controller, (c, d) the PI controller with coupled oscillators and (e, f) the FLC (Fuzzy). We performed simulations with fatigue and (a, c, e) $T_r = 30$ s and (b, d, f) $T_r = 60$ s.....	72
5.39	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are manually tuned, the previous PI controller tuned by GA (PI 1) and the new one (PI 2). (a) is the average quadriceps activation, and (b) is the average hamstrings activation.	73
5.40	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the average hamstrings activation, and (c) is the average gluteus activation.	74
5.41	Bar charts of the controllers activating quadriceps and hamstrings. The controllers are manually tuned, the previous PI controller tuned by GA (PI 1) and the new one (PI 2). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	75
5.42	Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.	76
5.43	Energy cost efficiency of FES cycling with (a) PI controller, (b) PI controller with coupled oscillators (PI+OSC), and (c) FLC (Fuzzy). The average energy cost efficiency was defined as the average total work over the average total metabolic energy cost (Section 4.5).	77
5.44	Metabolic energy cost, mechanical work rate and energy cost efficiency of FES cycling with (a) PI controller, (b) PI controller with coupled oscillators, and (c) FLC (Fuzzy). The energy is expressed in W/kg of muscle tissue.	78
6.1	Ergometer for rowing. We used the measures of this ergometer to develop the additional parts of the FES rowing model.	79

6.2	Rowing pose. The pose of the model had to be similar to the pose in the real system. Head-trunk and pelvis were fixed to simulate a person on a sitting position.....	81
6.3	Example of a simulation result. The top two graphs are the activation of the muscle groups. In (a), the third graph in is the angle ($^{\circ}$) of the motion. 0° is the Catch position, and 180° is the Handle Pull position. The blue line is the angle acquired from the data of OpenSim, and the green line is the output of the coupled oscillators. The fourth graph is the position (m) of the seat, and the fifth graph is the velocity (m/s) of the seat. In (b) the third graph is the force (N) which simulates the handle pulling. The fourth graph is the position (m) of the handle, and the fifth graph is the velocity (m/s) of the handle.....	84
6.4	Example of a bar chart of a performance measure. Each bar is the performance of a control strategy.....	84
6.5	Bar charts for controllers with constant activation: (a) the maximum velocity (m/s) and (b) the number of strokes per minute.....	85
6.6	FES rowing simulation results for controller with constant activation of (a, b) 0.25, (c, d) 0.50 and (e, f) 0.75. The graphs show the measures of (a, c, e) the seat and (b, d, f) the handle.....	86
6.7	Neuronal activities u_i (up), u_h (middle) and u_t (down) of the control layer. Initial inputs are $[S_i, S_t] = [1, 0]$, then it changes to $[0, 0]$, $[0, 1]$ and $[0, 0]$ again at 25, 50, and 75 iterations respectively.....	88
6.8	Outputs X (blue) and Y (red) of the timing layer. Initial inputs are $[S_i, S_t] = [1, 0]$, then it changes to $[0, 0]$, $[0, 1]$, $[0, 0]$ and $[1, 0]$ at 1.0, 5.0, 7.5 and 8.5 seconds respectively.....	88
6.9	System diagram for FES rowing with the controller based on attractor dynamics. The control layer is equivalent to the FSC and receives besides other inputs the values of S_i and S_t . The output is the neural activity of the three neurons, which activate the primitives in the timing layer. The timing layer is equivalent to the lower level controller. The activation of the muscles (U) is given by x (blue line).....	89
6.10	FES rowing simulation results for controller based on attractor dynamics. The graphs show the measures of (a) the seat and (b) the handle.....	90
6.11	Bar charts for controllers with constant activation of 0.5 (Constant), based on attractor dynamics (ATT), and based on attractor dynamics with amplitude control (ATT+AMP), with coupled oscillators (ATT+OSC) and both (ATT+OSC+AMP): (a) average quadriceps activation, (b) nonzero average quadriceps activation, (c) average hamstrings activation, and (d) nonzero average hamstrings activation.....	91
6.12	Bar charts for controllers with constant activation of 0.5 (Constant), based on attractor dynamics (ATT), and based on attractor dynamics with amplitude control (ATT+AMP), with coupled oscillators (ATT+OSC) and both (ATT+OSC+AMP): (a) the maximum velocity (m/s) and (b) the number of strokes per minute.....	92
6.13	FES rowing simulation results for controller based on attractor dynamics with amplitude control. The graphs show the measures of (a) the seat and (b) the handle.....	93
6.14	FES rowing simulation results for controller based on attractor dynamics with coupled oscillators. The graphs show the measures of (a) the seat and (b) the handle. θ_{OSC} corresponds to σ	95
6.15	FES rowing simulation results for controller based on attractor dynamics with coupled oscillators and amplitude control. The graphs show the measures of (a) the seat and (b) the handle. θ_{OSC} corresponds to σ	96
6.16	Energy cost efficiency of FES rowing with (a) controller based on attractor dynamics, (b) controller based on attractor dynamics with amplitude control, (c) controller based on attractor dynamics with coupled oscillators, and (d) controller based on attractor dynamics with coupled oscillators and amplitude control. The average energy cost efficiency was defined as the average total work over the average total metabolic energy cost (Section 4.5).....	97

6.17 Metabolic energy cost, mechanical work rate and energy cost efficiency of FES rowing with (a) controller based on attractor dynamics, (b) controller based on attractor dynamics with amplitude control, (c) controller based on attractor dynamics with coupled oscillators, and (d) controller based on attractor dynamics with coupled oscillators and amplitude control. The energy is expressed in W/kg of muscle tissue. 98

List of Tables

1	RMSE dos controladores PI, PI com osciladores acoplados (PI+OSC) e FLC para uma velocidade constante de 300 ⁰ /s. Foram considerados três escalas diferentes do modelo em simulações sem modificações do ambiente e com ruído. As simulações com carga não funcionaram corretamente para as escalas diferentes da original e por isso foram desconsideradas.	
2	Atividade neural da camada de controle baseada nos osciladores de Matsuoka.....	
2.1	Lower limbs joints of the human body. The last column specifies the type of the synovial joints (Marieb and Hoehn, 2007).	5
2.2	Shape os synovial joints, and which type of movement can occur (Marieb and Hoehn, 2007).....	5
2.3	Muscles related to the hip and knee joints, and the type of action they produce. Some muscles do almost the same action and are part of a muscle group. Hence, they were placed in the same cell of the table. The first name is the name of the muscle group (Marieb and Hoehn, 2007).	7
5.1	Initial angles and which joints are locked in the FES cycling model. The joint names follow the nomenclature from OpenSim standard.	26
5.2	The parameters for the controllers which used quadriceps and hamstrings muscle groups. K_p and K_i are the proportional and integrative constants of the PI control. The angle interval is the crank angles when a muscle is activated. RQ is right quadriceps, and RH is right hamstrings. The muscle groups of the left leg have an angle interval equal to the angle interval of the right leg delayed by 180°. The gain for $u < 0$ is the gain for the opposing muscle group activation. QH-Manual represents the manually tuned PI controller, and QH-PI 1 represents the PI controller tuned by GA with the same FSC of QH-Manual. QH-PI 2 represents the PI controller tuned by GA whose FSC used the muscle group of the opposing leg to decrease the velocity faster.	32
5.3	The parameters for QHG-PI and QHG-PID. K_p , K_i and K_d are the proportional, integrative and derivative constants of the PI control. The angle interval is the crank angles when a muscle is activated. RQ is right quadriceps, RH is right hamstrings, and RG is the right gluteus. The muscle groups of the left leg have an angle interval equal to the angle interval of the right leg delayed 180°. The gain for $u < 0$ is the gain for the opposing muscle group activation.....	43
5.4	Fuzzy rules for QHG-Fuzzy.....	43
5.5	Initial angles and which joints are locked in the FES cycling model which is 1.70 m tall.....	57
5.6	Initial angles and which joints are locked in the FES cycling model which is 1.90 m tall.....	58
6.1	Initial angles and which joints are locked in the FES rowing model. The joint names follow the nomenclature from OpenSim standard. cylinderToBase is the joint that connects the handle to the ergometer base.....	82
6.2	Neural activity based on Matsuoka oscillators for the three primitives of the model based on attractor dynamics.	87
6.3	The parameters for the controllers based on attractor dynamics. a is the gain of the attractors, b is the gain of the Hopf oscillator, q is the gain of the error, a_{c1} is the coupling constant, and ω_{ref} if the desired frequency of the oscillation. Attractor represents the controller based on attractors dynamics, and Attractors + Oscillators represents the controller based on attractors dynamics with the addition of coupled oscillators.	90

SYMBOLS LIST

Acronyms

CNS	Central nervous system
SCI	Spinal Cord injury
FES	Functional electrical stimulation
SCS	Spinal cord stimulation
ASIA	American Spinal Injury Association
BMI	Brain-machine interface
CPG	Central pattern generator
Vn4D	Visual Nastran 4D
AM	Amplitude modulation
PWM	Pulse width modulation
FLC	Fuzzy logic controller
PID	Proportional-integral-derivative controller
NN	Neural network
GA	Genetic algorithm
PSO	Particle swarm optimization
2D	Two dimensional
3D	Three dimensional
ODE	Ordinary differential equations
FSC	Finite state controller
RMSE	Root mean square error
QH- Manual	Manually tuned PI controller that activates quadriceps and hamstrings muscle groups
QH-PI 1	PI controller tuned by GA that activates quadriceps and hamstrings muscle groups
QH-PI 2	PI controller tuned by GA whose FSC used opposing muscle
QHG-PI	PI controller that activates quadriceps, hamstrings, and gluteus muscle groups
QHG-PID	PID controller that activates quadriceps, hamstrings, and gluteus muscle groups
QHG- Fuzzy	Fuzzy logic controller that activates quadriceps, hamstrings, and gluteus muscle groups
QHG- PI+OSC	PI controller with coupled oscillators
QHG- PI(1.90)	PI controller for the musculoskeletal model with 1.90 <i>m</i>
QHG- Fuzzy(1.90)	Fuzzy logic for the musculoskeletal model with 1.90 <i>m</i>
QHG-PI+ OSC(1.90)	PI controller with coupled oscillators for the musculoskeletal model with 1.90 <i>m</i>
QHG- PI(1.70)	PI controller for the musculoskeletal model with 1.70 <i>m</i>
QHG- Fuzzy(1.70)	Fuzzy logic for the musculoskeletal model with 1.70 <i>m</i>

QHG-PI+ OSC(1.70)	PI controller with coupled oscillators for the musculoskeletal model with 1.70 <i>m</i>
ATT	Controller based on attractors dynamics
ATT+AMP	Controller based on attractors dynamics with amplitude control
ATT+OSC	Controller based on attractors dynamics with coupled oscillators
ATT+AMP +OSC	Controller based on attractors dynamics with coupled oscillators and amplitude control

Latin Symbols

t	Time
k	Natural number
a	Constant parameter
b	Constant parameter
c	Constant parameter
a_1	Constant parameter of a nonuniform oscillator
a_2	Constant parameter of a nonuniform oscillator
r	Radius of the Hopf oscillator
\dot{r}	Derivative of the radius of the Hopf oscillator
m	Amplitude of the Hopf oscillator
x_i	Initial state
x_t	Target state
x	Output of the model based on attractor dynamics
\dot{x}	Derivative of the output of the model based on attractor dynamics
y	Auxiliar variable of the model based on attractor dynamics
\dot{y}	Derivative of the auxiliar variable of the model based on attractor dynamics
u_i	Activity level of the initial neuron
u_h	Activity level of the Hopf neuron
u_t	Activity level of the target neuron
\dot{u}_i	Derivative of the activity level of the initial neuron
\dot{u}_h	Derivative of the activity level of the Hopf neuron
\dot{u}_t	Derivative of the activity level of the target neuron
b_i	Input that activates the initial neuron
b_t	Input that activates the target neuron
U	Amplitude of muscle activation of the basic framework
I	Input vector of the basic framework
O_1	Output vector of the basic framework which is used to calculate the error
O_2	Output vector of the basic framework which is the input of the Finite State Controller block
P	Input vector of the basic framework which changes the environment
t_{r1}	The time to begin the transition of the ramp input
t_{r2}	The time to end the transition of the ramp input
f	Fatigue factor
\dot{f}	Derivative of the fatigue factor
T_f	Fatigue time constant
T_r	Resting time constant
u	Control signal
x^k	kth individual
$f(x^k)$	Fitness function for the kth individual
$f_n(x^k)$	Normalized fitness function for the kth individual
S	Total normalized <i>fitness</i> of the population
p^k	Probability of selection of the kth individual
q^k	Cumulative probability of the kth individual
N	Number of individuals/Number of samples
r_{nd}	Random number in the interval [0, 1]
\dot{E}	Rate of the muscle energy expenditure model
\dot{h}_{AM}	Activation and maintenance heat rate
\dot{h}_{SL}	Shortening/lengthening heat rate

\dot{w}	Mass specific mechanical work rate
\bar{w}	Average total work
\bar{E}	Average total metabolic energy
a_{c1}	Constant parameter of the controller oscillator
a_m	Constant parameter of the model oscillator
K_p	Proportional constant of the PID control
K_i	Integrative constant of the PID control
K_d	Derivative constant of the PID control
t_1	Time interval between 14 and 19 seconds
t_2	Time interval between 24 and 29 seconds
e	Error
e_i	Integration of the error as a function of time
Δe	Change in the error
G_e	Scaling factor of the error between the desired velocity and the crank velocity
$G_{\Delta e}$	Scaling factor of the change of error between the desired velocity and the crank velocity
s	Seat position
s_n	Seat position scaled to the interval $[-1, 1]$
v	Seat linear velocity
S_i	Input that activates the initial neuron
S_h	Input that activates the Hopf neuron
S_t	Input that activates the target neuron
t_r	Rise time constant of the neuron
t_a	Time lag of the adaptation effect of the neuron
w	Weight of inhibitory synaptic connection between neurons
x_i	Inner state of the initial neuron
x_h	Inner state of the Hopf neuron
x_t	Inner state of the target neuron
\dot{x}_i	Derivative of the inner state of the initial neuron
\dot{x}_h	Derivative of the inner state of the Hopf neuron
\dot{x}_t	Derivative of the inner state of the target neuron
v_i	Variable that represents the adaptation of the initial neuron
v_h	Variable that represents the adaptation of the Hopf neuron
v_t	Variable that represents the adaptation of the target neuron
\dot{v}_i	Derivative of the variable that represents the adaptation of the initial neuron
\dot{v}_h	Derivative of the variable that represents the adaptation of the Hopf neuron
\dot{v}_t	Derivative of the variable that represents the adaptation of the target neuron
s_{hp}	Handle Pull position
s_c	Catch position
q	Constant gain

Greek Symbols

θ	Angle
$\dot{\theta}$	Derivative of a angle
θ_0	Initial angle
π	Pi
ω	Angular velocity/Frequency
σ	Angle
$\dot{\sigma}$	Derivative of a angle
ϑ	Natural angular velocity/frequency of a constant oscillator
ϕ	Phase difference between two oscillators
$\dot{\phi}$	Derivative of the phase difference between two oscillators
μ	Natural angular velocity/frequency of the dimensionless differential equation
τ	Time multiplied by the constant parameter of a nonuniform oscillator
θ_1	Angle of a nonuniform oscillator
$\dot{\theta}_1$	Derivative of a nonuniform oscillator
ω_1	Natural angular velocity/frequency of a nonuniform oscillator
θ_2	Angle of a nonuniform oscillator
$\dot{\theta}_2$	Derivative of a nonuniform oscillator
ω_2	Natural angular velocity/frequency of a nonuniform oscillator
ω^*	Compromise frequency
μ_i	Constant parameter of the initial neuron
μ_h	Constant parameter of the Hopf neuron
μ_t	Constant parameter of the target neuron
ω_1	First angular velocity/frequency of the ramp input
ω_2	Second angular velocity/frequency of the ramp input
η	Energy cost efficiency
θ_{c1}	Angle of the controller oscillator
$\dot{\theta}_{c1}$	Derivative of the controller oscillator
ω_{c1}	Natural angular velocity/frequency of the controller oscillator
θ_m	Angle of the model oscillator
$\dot{\theta}_m$	Derivative of the model oscillator
ω_m	Natural angular velocity/frequency of the model oscillator
ω_{ref}	Desired angular velocity/frequency
θ_{c2}	Angle of the controller oscillator with constant frequency
γ	Muscle activation
$\bar{\gamma}$	Average muscle activation
β	Parameter that determines the steady-state firing rate for a constant input

1

INTRODUCTION

1.1 CONTEXT

The central nervous system (CNS) of the human body communicates with the other systems below the neck through the spinal cord. It can send commands to regulate autonomous functions (e.g., heartbeat, breathing) or control limb movement (Marieb and Hoehn, 2007). A large number of sensors spread all over the body also send signals to the spinal cord.

The spinal cord injury (SCI) disables the communication between higher structures of CNS and other systems (Popović and Sinkjær, 2012). The most notable affected functions are the voluntary movements and sensations from parts of the body innervated by neurons below the lesion. If the SCI is incomplete, the person can still sense the body and move the limbs, even if just a few. However, if the SCI is complete, these functionalities are completely lost. The affected parts of the body depend on the location of the injury. If the damage is in the highest part of the spinal cord (cervical spine), all parts of the body below the neck can be affected, especially the limbs. If the injury is below the thoracic spine, the most affected parts of the body are the lower limbs. The first is known as tetraplegia, and the late is known as paraplegia.

Due to lack of proper neural activity, other effects normally occur, such as drop of muscle mass and decrease in bone density (Stein, 1999). Additionally, people with SCI can present a slow healing because of low blood circulation, as well as poor control of body temperature and other autonomous functions. The sedentary lifestyle and the weakening of the body have increased the number of deaths of people with SCI, especially due to cardiovascular diseases (Washburn and Figoni, 1999).

Activities based on functional electrical stimulation (FES) are helping to reduce the aforementioned effects and to improve some cardiovascular parameters, such as heart rate, stroke volume and blood pressure (Deley et al., 2014). FES consists in the application of small current electrical pulses to nerves mostly via electrodes, which can be invasive or noninvasive. Invasive electrodes are a quite expensive method and often also risky (Loeb et al., 2006), so noninvasive electrodes are more common in FES applications. When the electrodes are placed near skeletal muscles, the nerves affected by the electric pulses transmit the generated action potentials to the muscles, which contract. As a consequence, it is possible to generate movement even if the person cannot voluntarily contract the muscles affected by the lesion. The common FES applications, which are focused on the movement of lower limbs, are standing, walking and cycling, but other applications exist, like rowing. With the improvement of their physiological condition, people with SCI can achieve a better quality of life, including the reduction of the incidence of related diseases.

A brief comparison between each application and their relevance to SCI would lead to conclude that FES standing and walking often require excessive effort to perform the desired action (Kralj and Bajd, 1989). Besides, it is necessary to support the body (e.g., sidebars, holders) to prevent falling and reduce the burden on the legs. On the other hand, even though FES cycling evokes lower muscle forces compared to standing and walking (Fornusek et al., 2013), it may be simpler and safer. Also, it generates a lot of other benefits for the physically disabled (Berkelmans, 2008). FES rowing goes further and combines the stimulation of paralyzed leg muscles with voluntary upper body exercise, which increases cardiorespiratory benefits compared to FES applications alone (Ektas et al., 2014).

One additional feature that characterizes FES cycling and rowing is that both are rhythmic movements. A rhythmic movement consists of a pattern which repeats over a period. Hence, it is possible to use specific strategies for this kind of motion. There are studies about vertebrate animals that evidence the presence of modular motion generators in the spinal level called motor primitives (see review by (Degallier and Ijspeert,

2010)). Among these motor primitives, some are networks composed of multiple coupled oscillatory centers which control rhythmic motions: the central pattern generators (CPGs).

Even though individuals with complete and chronic lesions usually have a low probability of recovering, some studies showed that it is possible to enhance the excitability and facilitate plasticity of the neural structures in the damaged area if there are still available neural pathways. These studies used especially spinal cord stimulation (SCS) based on epidural stimulation (i.e., stimulation with electrodes invasively inserted in the spinal chord) (Harkema et al., 2011; Angeli et al., 2014), and transcutaneous stimulation (i.e., stimulation with electrodes externally placed on the spinal chord) (Gerasimenko et al., 2015). The tests were accomplished with participants with SCI clinically classified as ASIA (American Spinal Injury Association) impairment scale A or B. All subjects obtained a limited but persistent motor recovery even after the end of the protocol. There is also a study based on a brain-machine interface (BMI), proprioceptive feedback on the upper limb, and robotic gait training that also reported function recovery on individuals with complete SCI (Donati et al., 2016).

Inspired by those recent results, this work is part of a broader research effort targeting SCI rehabilitation based on principles of neuro-engineering and robotics integrated with goal-directed physical therapy. As a first step, the selected participants with SCI will pass through a physical conditioning protocol. We intend to use an adapted tricycle and a rowing machine available to perform FES exercises.

1.2 PROBLEM DEFINITION

Tests with a real tricycle and a rowing machine are time-consuming and need many preparations. Besides the development and instrumentation of the equipment, a team has to perform the system set up and check if the safety measures are good enough. Additionally, the participants must be accompanied by a clinician and displace to the test location. Usually, a session for FES cycling or rowing where 30 to 60 minutes of exercise are performed would involve roughly 30 additional minutes of initialization and conclusion. The available time for experiments that may lead to improved systems may be reduced due to weak muscles, spasms, and the need to rest due to fatigue induced by stimulation (Hunt et al., 2002; Berkelmans, 2008; Fornusek et al., 2013).

In this scenario, designing new controllers for FES cycling and rowing applications may prove a challenging task, particularly considering controllers based on motor primitives and CPGs, which may require several sessions for parameter tuning. The complexity may increase if these controllers are combined with conventional techniques, such as PID.

These controllers must also be designed to provide satisfactory performance for different subjects and conditions. Hence, an additional challenge concerns finding an appropriate method to compare these control systems.

One potential alternative for both issues involve an intense and structured use of numerical simulation to support the development of FES control systems. This approach is explored in this work.

1.3 OBJECTIVES

The main objective of this work concerns the design and evaluation in simulation of new controllers for rhythmic FES-assisted exercises, such as cycling and rowing. The additional specific goals are defined as follows:

- Development of a simulation framework for FES applications including detailed musculoskeletal models;

- Use of optimization methods to tune the parameters of the controllers;
- Conception of controllers that explore the properties of rhythmic movements, hence possibly enabling application on other rhythmic movements;
- Proposal of methods to compare the performance of different FES-aided exercises;
- Comparison of performance of different controllers in different user and environment configurations.

1.4 MANUSCRIPT ORGANIZATION

This work is organized as follows. The next section describes some theoretical foundations for a better comprehension of the text. Chapter 3 presents state of the art in FES cycling and FES rowing. The general methodology is given in chapter 4, including the method to develop controllers, as well as the configuration of the simulation environment.

Following, chapter 5 describes the developed work for FES cycling. First, it remarks the main characteristics of the developed musculoskeletal model. The computer simulations are presented in the next section. The simulations were divided according to the type of controller and environment, and the subsections include the specific methodology for that type of simulation, results and a brief analysis. The chapter concludes with the required results for the comparison between FES cycling and FES rowing. Chapter 6 is organized like chapter 5 but for FES rowing.

Finally, on chapter 7 the general analysis about the work is presented, and then the last chapter presents the concluding remarks and future works.

2

THEORETICAL FOUNDATIONS

2.1 MUSCULOSKELETAL SYSTEM

The human body can perform a large number of complex movements due to the combination of functionalities and structures of the body, especially bones, joints, and muscles. When studying the human body, it is important to define anatomical planes and directions (Marieb and Hoehn, 2007). They will help locate the parts of the body and understand the types of movements the body can perform. The body is frequently sectioned using the following planes:

- Sagittal plane: a vertical plane that divides the body into left and right halves;
- Frontal plane: also a vertical plane, but divides the body into anterior (front) and posterior (back) parts;
- Transversal plane: a horizontal plane that divides the body into superior and inferior parts.

Since this work concentrates on lower limbs, this section will focus on the systems of the inferior part of the body.

Starting from the skeletal system, the bones we are interested are part of the appendicular skeleton, i.e., the bones of the limbs and the ones which attach the limbs to the rest of the body (Marieb and Hoehn, 2007). The hip girdle is responsible for attaching the bones of the lower limbs to the vertebral column. It also transmits the weight of the upper body to the lower limbs and upholds the visceral organs of the pelvis. The bones of the lower limbs are divided into three segments: thigh, leg, and foot. The bone of the thigh is the femur, which is the longest and strongest bone in the body. The leg has two parallel bones, the tibia, and fibula. These bones have almost no movement between them, but are strong and help to stabilize the limb. The bones of the foot are responsible for supporting the body weight. Additionally, they help to propel the body forward during the walk and run. The bones of the foot are highly segmented and are divided into tarsus (posterior part of the foot), metatarsus and phalanges (toes).

Besides the mentioned bones of the lower limbs, there is still one bone located in the knee, the patella. It is a bone enclosed in the tendon that secures the anterior thigh muscles to the tibia. It acts as a lever of the thigh muscles and also protects the knee joint. In summary, the lower limbs have a total of 60 bones plus 2 bones of the hip girdle.

As mentioned above, the knee, which is the place where the femur meets the tibia, fibula, and patella, is a joint. It is because of the joints that the skeleton can move and is held together (Marieb and Hoehn, 2007). Structurally, they can be classified as:

- Fibrous joint: mostly immovable, it has dense fibrous connective tissue and no joint cavity;
- Cartilaginous joints: it is slightly movable, and is composed of cartilage. Also, does not have joint cavity;
- Synovial Joint: freely movable, it has a fluid-containing cavity.

The types of the lower limbs joints are shown in Table 2.1. It is possible to see that almost all joints of the lower limbs are synovial, and that allows a high range of movements. Each synovial joint has a specific shape, which defines the type of movement that can occur. The compilation of the shapes and movements are in Table 2.2.

The knee joint is the largest and one of the most complex joints in the human body (Marieb and Hoehn, 2007). The tibiofemoral joint that is shown in Table 2.1 actually consists of two joints. Hence, the knee is the

Table 2.1: Lower limbs joints of the human body. The last column specifies the type of the synovial joints (Marieb and Hoehn, 2007).

Joint	Articulation bones	Structural type	Synovial joint type
Hip	Hip bone; Femur	Synovial	Ball-and-Socket
Knee tibiofemoral	Femur; Tibia	Synovial	Modified hinge
Knee femoropatellar	Femur; Patella	Synovial	Plane
Superior tibiofibular	Tibia; Fibula	Synovial	Plane
Inferior tibiofibular	Tibia; Fibula	Fibrous	-
Ankle	Tibia and fibula; Talus (Tarsal bone)	Synovial	Hinge
Intertarsal	Adjacent tarsals	Synovial	Plane
Tarsometatarsal	Tarsal(s); Metatarsal(s);	Synovial	Plane
Metatarsophalangeal	Metatarsal(s); Phalanx	Synovial	Condylar
Interphalangeal	Adjacent phalanges	Synovial	Hinge

Table 2.2: Shape of synovial joints, and which type of movement can occur (Marieb and Hoehn, 2007).

Synovial joint type	Shape	Movement
Plane	Flat articular surfaces	Gliding
Hinge	Cylinder-Trough surfaces	Flexion and extension
Pivot	Sleeve-Axle surfaces	Rotation
Condylar (Ellipsoid)	Oval articular surfaces	Flexion and extension
		Adduction and abduction
Saddle	Concave-Convex articular surfaces	Flexion and extension
		Adduction and abduction
Ball-and-Socket	Spherical Head-Cup surfaces	Flexion and extension
		Adduction and abduction
		Rotation

junction of three joints. It acts primarily as a hinge joint, but structurally it is a bicondylar joint. Hence, some rotation is possible when the knee is partially flexed, and when the knee is extending. The femoropatellar joint is a plane joint, and the patella glides across the inferior end of the femur during knee flexion. Additionally, the complex shape of the joint provides a steady support for the body while standing.

There are a great variety of movements that can occur in the joints. Table 2.2 shows some of these movements. Gliding occurs when one flat, or nearly flat, bone surface slips over another. Rotation, as the name implies, is a rotation along the axis of a bone connected to the joint. Flexion is the action of bringing the articulating bones closer to each other, usually along the sagittal plane. Abduction is the action of move a limb away from the middle of the body along the frontal plane. Extension and adduction are the opposite movements of flexion and abduction, respectively.

All voluntary movements of the human body are a result of muscle contraction (Marieb and Hoehn, 2007). In the case of skeletal muscles, they are attached and covering the bones, and they are responsible for the movement of the joints. Skeletal muscles are very adaptable. They can apply a small and constant force to grab a plastic cup or a large force with a quick response to jump.

The muscle can perform movements because they transform chemical energy into mechanical energy. It starts with a signal that the nervous system sends to control the muscle activity. In general, the neurons connect to the muscles in the central part of the muscles. In the neuromuscular junction, Na^+ diffuses in, and K^+ diffuses out, causing a depolarization of the sarcolemma (i.e., the plasma membrane of the muscle cell). As a

consequence, an action potential is ignited, and it propagates in all directions of the sarcolemma. The impulses are conducted to the deepest parts of the muscle cell through T-tubules, which are elongated protuberances of the sarcolemma. Deep in the muscle cell, the action potential indirectly contracts the myofilaments, i.e., the structures responsible for the contraction of the muscle cell. The contraction is triggered by the release of Ca^{2+} in the cytosol, and a series of events in the myofilaments called Cross-Bridge Cycle.

Even though we can associate “contraction” with “shortening,” the contraction refers only to the activation of the cross-bridges (Marieb and Hoehn, 2007). It is possible to generate muscle contraction without movement, which is called isometric contraction. The velocity and duration of contraction will depend on the muscle fiber (cell) type. They can be classified according to the speed of contraction. The slow fibers contract slowly, but they have high endurance and resistance to fatigue. The fast fibers contract rapidly, but they also tire rapidly and are fatigable. Hence, slow fibers are best suited to endurance-type exercises, while fast fibers are best suited for short-term, intense exercises.

It is still not clear the cause of fatigue, but ionic imbalances in the muscle cell contribute to its occurrence. Muscle fatigue is characterized by the physiological inability to contract even if the muscle is being stimulated. In summary, short-term, intense exercises cause a rapid fatigue, but the recovery is also fast. On the other hand, long-term with low-intensity exercises develop fatigue slowly and require a long time to recover from the fatigue completely.

Muscles can be classified into four functional groups (Marieb and Hoehn, 2007):

- Prime mover: major muscle that produces a specific movement;
- Antagonist: muscle that opposes the movement produced by the prime mover;
- Synergist: muscle that adds an extra force to the movement or reduces undesired movements;
- Fixator: synergist muscle that immobilizes a bone.

One crucial characteristic of antagonists is that they also help to regulate the actuation of a prime mover. They contract slightly to provide some resistance, preventing overshooting or even stopping the movement. When a prime mover and its corresponding antagonist apply the same force to a joint, the joint does not move, and the contraction is isometric.

Since the developed controllers of this project act in the muscles crossing the hip and knee joints, Table 2.3 shows the muscles related to these joints and the type of action they produce.

2.1.1 OpenSim

OpenSim is an open-source software for development of musculoskeletal models and dynamic simulations (Delp et al., 2007). It was developed to exchange, analyze and improve simulations through a multi-institutional collaboration. Applications range from orthoses development to analysis of athletic performance. Some useful features from OpenSim are the Analyze Tool and Probes. They report information from simulations like actuators forces, internal loads, and even metabolic cost. There is the option to interface OpenSim and Matlab, and use Matlab tools to analyze simulations.

Most of the 3D musculoskeletal models in state of the art were developed using Visual Nastran 4D (Vn4D) (Abdulla and Tokhi, 2014; Yahaya et al., 2014; Hussain et al., 2008). The body is mostly made of geometrical parts with standard anthropometric dimensions, and the users provide the muscle model. It depends on the purpose of the model, but in general, the musculoskeletal models from OpenSim have a high number of bones, joints, and muscles. The geometry of the bones helps to define the center of mass. Also, the muscles are

Table 2.3: Muscles related to the hip and knee joints, and the type of action they produce. Some muscles do almost the same action and are part of a muscle group. Hence, they were placed in the same cell of the table. The first name is the name of the muscle group (Marieb and Hoehn, 2007).

Muscles/Muscle groups	Type/Location	Action
Iliopsoas - Iliacus - Psoas major	Anterior muscles	Prime mover for flexing thigh. Prime mover for flexing thigh. Also flexes vertebral column laterally.
Sartorius	Adductor muscle	Flexes, abducts, and laterally rotates thigh.
Adductors - Adductor magnus - Adductor longus - Adductor brevis	Adductor muscles	Adducts and medially rotates and flexes thigh. Also, extend thigh. Adducts, flexes, and medially rotates thigh. Adducts, flexes, and medially rotates thigh.
Pectineus	Adductor muscle	Adducts, flexes, and medially rotates thigh.
Gracilis	Adductor muscle	Adducts thigh, flexes and medially rotates leg.
Quadriceps femoris - Rectus femoris - Vastus lateralis - Vastus medialis - Vastus intermedius	Anterior muscles	Extends knee and flexes thigh at hip. Extends and stabilizes knee. Extends knee. Extends knee.
Tensor fasciae latae	Anterior muscle	Steadies the knee and trunk on thigh by making iliotibial tract taut.
Gluteus maximus	Posterior muscle	Major extensor of thigh.
Gluteus medius	Posterior muscle	Abducts and medially rotates thigh.
Gluteus minimus	Posterior muscle	Abducts and medially rotates thigh.
Piriformis	Posterior muscle	Rotates extended thigh laterally.
Obturator externus	Posterior muscle	Rotates extended thigh laterally.
Obturator internus	Posterior muscle	Rotates extended thigh laterally.
Gemellus	Posterior muscle	Rotates extended thigh laterally.
Quadratus femoris	Posterior muscle	Rotates thigh laterally and stabilizes hip joint.
Hamstrings - Biceps femoris - Semitendinosus - Semimembranosus	Posterior muscles	Extends thigh and flexes knee. Extends thigh and flexes knee. Extends thigh and flexes knee.

connected to the bones at specific positions, and a specific contact surface generates a more realist contact with the ground and equipment.

A relevant feature of OpenSim musculoskeletal models is the knee joint. 3D models of the human body from studies found in state of the art adopt a revolute joint to simulate the knee joint. OpenSim models also use a single-degree-of-freedom model, but it accounts for the kinematics of both the tibiofemoral joint and the femoropatellar joint in the sagittal plane as well as the patellar levering mechanism (Yamaguchi and Zajac, 1989). Additionally, the adopted model also specifies the transformations between the femoral, tibial, and patellar reference frames as functions of the knee angle (Delp et al., 1990). Hence, OpenSim models are more suitable to compare simulation results with data from real tests.

2.2 FUNCTIONAL ELECTRICAL STIMULATION

Even though the muscles connected to the CNS below the level of a complete SCI are not controllable volitionally, the nerves which serve the muscles are still intact. Hence, it is possible to stimulate the neuromuscular tissues using an external source. Functional electrical stimulation (FES) consists in the application of small current electric pulses in the nerves which serve the muscles. It is also possible to use FES to interface with the sensory system (Popović and Sinkjær, 2012).

Generally, the stimulator delivers trains of short rectangular pulses. The rise time of a rectangular pulse is fast enough so that the membranes in the neuromuscular junction do not fail to open their channels. The stimulator tries to mimic the natural flow of excitation signals generated by the CNS, which delivers a sequence of pulses to keep the muscle contracting. If the stimulus is large enough, an action potential will be triggered, and it will propagate in both directions in the nerve. Generally, FES applications use short pulses with a period less than 200 μ s.

The interface with the body is done through capacitive conductors known as electrodes. It is most common the use of noninvasive electrodes, but it is also possible to use invasive electrodes (Loeb et al., 2006). However, invasive electrodes are quite expensive and need surgery. There are two possible configurations for the interface:

- Monopolar - An active electrode is placed near the neuro-muscular systems to be stimulated, and a single common electrode is positioned relatively distant to them;
- Bipolar - Two electrodes are positioned around the neuro-muscular system to be stimulated, closing the electrical circuit.

Most stimulators use current regulation because the control of the charge delivered to the tissue is more precise than voltage regulation (Popović and Sinkjær, 2012). However, the surface area of the electrodes must have to be chosen carefully. Large surface areas diffuse the current and may not trigger the action potentials. On the other hand, small surface areas may result in a high charge density and current density. Additionally, other properties as amplitude and width (duration) of the pulse may affect the electrical charge that will be delivered to the neuro-muscular structure. The parameters have to be chosen so as to prevent potential damage to the tissue and potential degradation of the electrode.

The recruitment of muscle fiber can be controlled with amplitude modulation (AM) or pulse width modulation (PWM). PWM is most used since it is easy to construct and has a good resolution (Popović and Sinkjær, 2012). When the CNS delivers a signal to recruit the muscle fibers, it first recruits slow fibers, which contract slowly but have high endurance and resistance to fatigue. Then it recruits fast fibers, which contract rapidly but also tire rapidly and are fatigable. The recruitment is typically the inverse when applying FES (Popović and Sinkjær, 2012). Hence, participants with SCI tire rapidly during sessions with FES and need to rest more

frequently than a participant who performs the exercise without FES.

2.3 COUPLED OSCILLATORS

2.3.1 Definition of oscillator

An oscillator is a phenomenon that describes a rhythmic behavior (see review by (Strogatz, 1994)). It may be defined as

$$\dot{\theta} = f(\theta), \quad (2.1)$$

and its solution is

$$\theta = f(\theta)t + \theta_0, \quad (2.2)$$

where θ is the position of a point on the circle, and $\dot{\theta}$ is the velocity (frequency) vector of this point. θ_0 is the initial position of the point, and t is the time. An important property of oscillators is that the angle is the same for $\theta = \theta_0 + 2\pi k$, $k = 0, 1, 2$. That means $f(\theta)$ is a periodic function.

For a simple oscillator, only the phase (i.e., θ) changes. There is no variation of amplitude (i.e., the circle radius). The simplest oscillator is the one with constant frequency, i. e.,

$$\dot{\theta} = \omega. \quad (2.3)$$

Nonuniform oscillators may be defined (Strogatz, 1994) using

$$\dot{\theta} = \omega + a \sin(\theta), \quad (2.4)$$

where a is a constant parameter, and ω is the natural frequency of the system. The relation between phase and frequency of a nonuniform oscillator is shown in Figure 2.1. For convenience, it is assumed that $\omega > 0$ rad/s and $a \geq 0$ rad/s. Note that Equation 2.4 is equal to Equation 2.3 when $a = 0$ rad/s. The properties of the oscillator change based on the value of a . The cases for $a < \omega$, $a = \omega$, and $a > \omega$ are shown in Figure 2.2. Note that there is a saddle point when $a = \omega$. The frequency at this point is 0 rad/s. For $a > \omega$, there are an attractor point and a repulsor point. All trajectories are attracted to the attractor point when t tends to infinity (Strogatz, 1994).

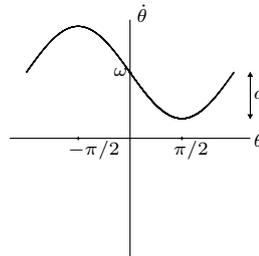


Figure 2.1: Relation between the angle (θ) and frequency ($\dot{\theta}$) of a nonuniform oscillator. The θ -axis is showing the interval $[-\pi, \pi]$ since the $\dot{\theta} = f(\theta)$ is a periodic function. The function oscillates around ω with amplitude a .

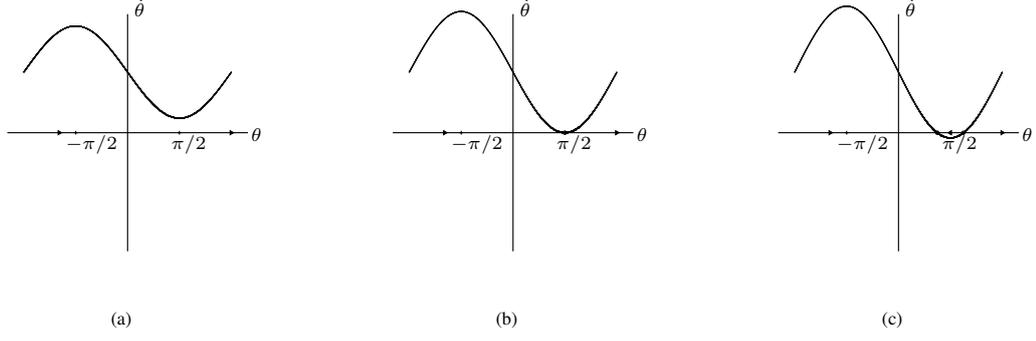


Figure 2.2: Relation between the angle (θ) and frequency ($\dot{\theta}$) of a nonuniform oscillator when (a) $a < \omega$, (b) $a = \omega$ and (c) $a > \omega$. There is a saddle point when $a = \omega$, and for $a > \omega$, there are an attractor point and a repulsor point. The arrows indicate the direction of θ when t increases.

2.3.2 Coupling oscillators

Consider a system model as a simple oscillator and a controller as a nonuniform oscillator. Their differential equations are, respectively,

$$\dot{\sigma} = \vartheta, \quad (2.5)$$

$$\dot{\theta} = \omega + a \sin(\sigma - \theta), \quad (2.6)$$

where the term $a \sin(\sigma - \theta)$ modifies the controller frequency, increasing or decreasing $\dot{\theta}$ until it is equal to ϑ . When both oscillators operate at the same frequency, the phase difference ($\sigma - \theta$) becomes constant, and then $\dot{\theta}$ stops changing. Therefore, the oscillators are coupled (Strogatz, 1994). The dynamics of the coupled oscillators is given by the differential equation of $\phi = \sigma - \theta$, which is

$$\dot{\phi} = \dot{\sigma} - \dot{\theta} = (\vartheta - \omega) - a \sin(\phi). \quad (2.7)$$

Note that Equation 2.7 is a nonuniform oscillator. To simplify the analysis, consider the dimensionless equation

$$\dot{\phi} = \mu - \sin(\phi), \quad (2.8)$$

where

$$\dot{\phi} = \frac{d\phi}{d\tau}, \quad (2.9)$$

$$\tau = at, \quad (2.10)$$

$$\mu = \frac{\vartheta - \omega}{a}. \quad (2.11)$$

The cases for $\mu = 0$, $0 < \mu < 1$ and $\mu > 1$ are shown in Figure 2.3.

Note that the attractor point is at $\phi = 0$ when $\mu = 0$, i.e., $\omega = \vartheta$. Hence, phase difference between the controller and the model is null. When $0 < \mu < 1$, $\dot{\phi} = \dot{\sigma} - \dot{\theta} = 0$ at the attractor point. In this case, controller and model have the same frequency, but there is a constant phase difference between them (i.e., the system is phase-locked). For $\mu = 1$, there is one saddle point. The phase difference indefinitely increases when $\mu > 1$ since there is no attractor point. Hence, the coupling will occur only when

$$\omega - a \leq \vartheta \leq \omega + a, \quad (2.12)$$

$$|\vartheta - \omega| \leq a. \quad (2.13)$$

In other words, the coupling only happens if the curves of Figure 2.3 cross the ϕ -axis.

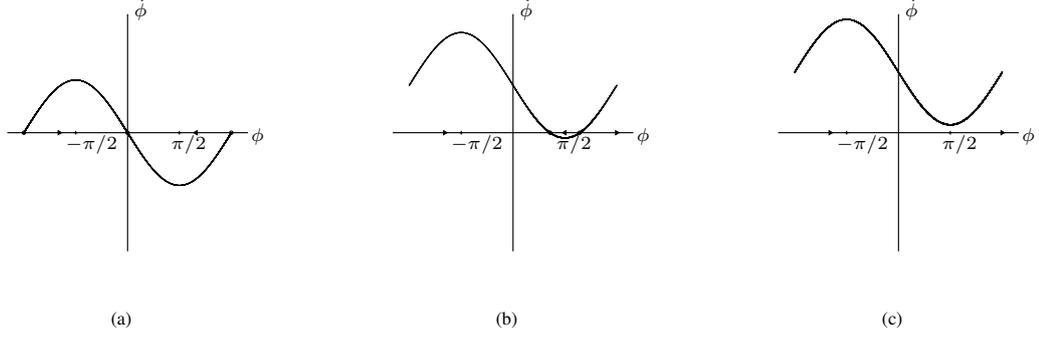


Figure 2.3: Relation between dimensionless angle (ϕ) and frequency ($\dot{\phi}$) of a nonuniform oscillator when (a) $\mu = 0$, (b) $0 < \mu < 1$ and (c) $\mu > 1$. For $\mu < 1$, there are an attractor point and a repulsor point. The arrows indicate the direction of ϕ when t increases.

If the model is also a nonuniform oscillator, the new differential equations are

$$\dot{\theta}_1 = \omega_1 + a_1 \sin(\theta_2 - \theta_1), \quad (2.14)$$

$$\dot{\theta}_2 = \omega_2 + a_2 \sin(\theta_1 - \theta_2), \quad (2.15)$$

where $\omega_1, \omega_2 > 0$ rad/s and $a_1, a_2 > 0$ rad/s. The system dynamics can be analyzed using its phase difference and frequency, i. e.,

$$\phi = \theta_1 - \theta_2, \quad (2.16)$$

$$\dot{\phi} = \dot{\theta}_1 - \dot{\theta}_2 = \omega_1 - \omega_2 - (a_1 + a_2) \sin(\theta_1 - \theta_2). \quad (2.17)$$

Note that the phase difference is also a nonuniform oscillator, and it has an attractor point and a repulsor point when $|\omega_1 - \omega_2| \leq a_1 + a_2$. A saddle point appears when $|\omega_1 - \omega_2| = a_1 + a_2$. $\dot{\phi} = \dot{\theta}_1 - \dot{\theta}_2 = 0$ rad/s at the attractor point, i.e., $\omega^* = \dot{\theta}_1 = \dot{\theta}_2$. ω^* is called compromise frequency, and it is between ω_1 and ω_2 . However, ω^* is not necessarily equidistant from ω_1 and ω_2 . Its value will depend on a_1 and a_2 .

2.4 MOTOR PRIMITIVES

Motor primitive is defined as a neural network of the CNS that activates a set of muscles to perform a movement (Degallier and Ijspeert, 2010). According to that definition, a combination of complex movements is then the summation of a finite, discrete set of motor primitives. This modular approach may be more straightforward when compared to a direct control of a high number of muscles since the high-level control only has to produce the signal for each motor primitive. Furthermore, motor primitives can represent discrete movements (i.e., movement from one pose to another), rhythmic movements and the combination of them.

Even though some movements have a pattern that repeats over time, they cannot be classified as a pure oscillation. Small variations of the movement and changes in the environment may affect the actual trajectory, so it is not possible to perform a fully rhythmic movement. Also, a task that has to perform a cyclical motion is finite in time, and the initial and final accelerations are zero.

The work of (van Mourik and Beek, 2004) concludes that the kinematic properties of the first and last half-cycles of a rhythmic motion are closer to those of discrete movements. Hence, a rhythmic movement in a finite time is a sequence of a discrete movement, a rhythmic movement and terminates with a discrete movement. (Schöner and Santos, 2001) developed a model based on attractor dynamics according to this hypothesis. Additionally, according to the definition of planning/execution phases (Degallier and Ijspeert,

2010), the generation of the motion follows the One/One hypothesis. The processes to choose the features of the movement (control layer) and the process responsible for the spatiotemporal activation of the muscles (timing layer) are the same for both types of movements.

The timing layer is modeled using three motor primitives: two attractors consisting of the initial state (x_i) and target state (x_t), and a Hopf oscillator of amplitude \sqrt{m} and frequency ω . If an attractor is active, the system stays at the same position when a perturbation occurs. A Hopf oscillator is a simple polynomial equation containing the bifurcation of a limit cycle at \sqrt{m} , which produces a stable periodic solution (Perko, 2013). The Hopf oscillator is expressed in Polar coordinates as

$$\dot{r} = mr - r^3, \quad (2.18)$$

$$\dot{\theta} = \omega. \quad (2.19)$$

Rewriting in Cartesian coordinates, it follows that

$$\begin{aligned} \dot{x} &= \dot{r}\cos\theta - r\dot{\theta}\sin\theta \\ &= (m - r^2)x - \omega y, \end{aligned} \quad (2.20)$$

and similarly

$$\dot{y} = (m - r^2)y + \omega x. \quad (2.21)$$

The equations of the timing layer are:

$$\dot{x} = -a|u_i|(x - x_i) + |u_h|(b(m - r^2)x - \omega y) - a|u_t|(x - x_t), \quad (2.22)$$

$$\dot{y} = -a|u_i|y + |u_h|(b(m - r^2)y + \omega x) - a|u_t|y, \quad (2.23)$$

where x is the output signal, y is an auxiliary variable, $r^2 = x^2 + y^2$ and $m = 1$. a and b control the convergence speed. i represents the attractor of the initial state, t represents the attractor of the target state, and h represents the Hopf oscillator. The primitives are multiplied by the activity level $|u_k|$ ($k = i, h, t$) of three neurons that are never fully active simultaneously. These neurons are part of the control layer, and the neuron activities are given by the equations

$$\alpha\dot{u}_i = \mu_i u_i - |\mu_i|u_i^3 - c(u_h^2 + u_t^2)u_i, \quad (2.24)$$

$$\alpha\dot{u}_h = \mu_h u_h - |\mu_h|u_h^3 - c(u_i^2 + u_t^2)u_h, \quad (2.25)$$

$$\alpha\dot{u}_t = \mu_t u_t - |\mu_t|u_t^3 - c(u_i^2 + u_h^2)u_t. \quad (2.26)$$

The first two terms of each equation are the normal form of a degenerate pitchfork bifurcation (Strogatz, 1994). There is an attractor at $u_k = 0$ for any value of μ_k , ($k = i, h, t$). However, the system becomes unstable when $\mu_k > 0$, and two new attractors at $u_k = -1$ and $u_k = 1$ appear. It is possible to use the value of μ_k to control the neuron activity. The attractors $+1$ and -1 are equivalent to the active state, and 0 is the inactive state (Schöner and Santos, 2001). The third term is a competitive term to ensure only one neuron is active at a time. The parameters μ_k are expressed as

$$\mu_i = 1.5 + 2b_i, \quad (2.27)$$

$$\mu_h = 1.5 + 2(1 - b_i)(1 - b_t), \quad (2.28)$$

$$\mu_t = 1.5 + 2b_t, \quad (2.29)$$

where b_i and b_t are the inputs of the control layer. An example of rhythmic motion is shown in Figure 2.4.a. The system starts with $b_i = 1$ and $b_t = 0$, i.e., it starts at the initial state. When b_i is set to 0, the oscillation begins. Then b_t is set to 1 to inhibit the oscillation and finish the movement at the target state. In the case of

a discrete movement, the movement ends after a half-cycle. However, if b_t works as a boolean variable, the oscillations can end in the middle or after the half-cycle. An example of b_t working as a boolean variable is shown in Figure 2.4.b. A better approach is to use the equation

$$b_t = 1 - \frac{\tanh(10(0.7x_t - x)) + 1}{2}, \quad (2.30)$$

which autonomously turns the oscillatory state off when x is close to x_t . An example of the use of Equation 2.30 is shown in Figure 2.4.c.

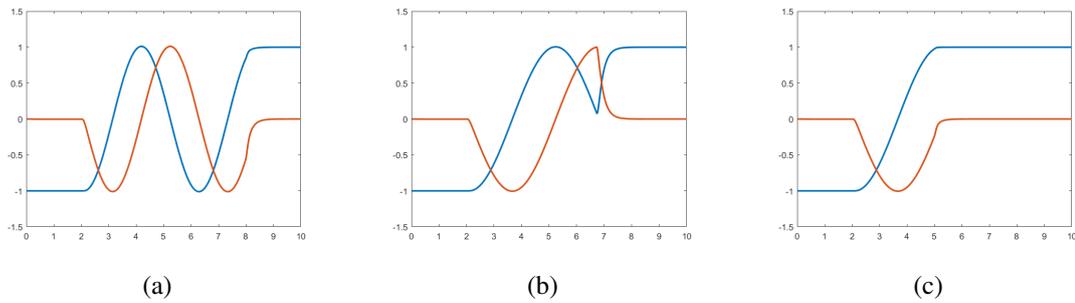


Figure 2.4: Outputs X (blue) and Y (red) of the time layer. (a) is an example of rhythmic motion. (b) is an example of discrete motion with b_t working as a boolean variable. (c) is an example of discrete motion using the Equation 2.30. The initial state and target state are -1 and 1 , respectively. The amplitude of the oscillation is 1 .

3

STATE OF THE ART

FES has been used to improve quality of life and restore lost functions of the population with paraplegia and other neuromuscular disorders (Gföhler and Lugner, 2004; Hunt et al., 2012). For instance, FES has been used for muscle strengthening, correct drop-foot, and as a functional orthosis. It has also been used for aerobic exercises (Chen et al., 2004).

This chapter presents what has been done with FES cycling and rowing, focusing especially on the development of control strategies. The last section focuses on the use of computer simulations to develop control strategies for FES applications.

3.1 FES CYCLING

More than 30 years have passed since research groups started to study cycling assisted by FES (Glaser et al., 1983; Petrofsky et al., 1983; Eichhorn et al., 1984). FES cycling is simpler and safer than standing and walking. Its methods control knee flexion and extension in the sagittal plane to move a crank connected to the wheels while the participant is sitting. Also, the participant uses an orthosis to fix the angle of the ankle and reduce the number of degrees of freedom. Hence, there is no risk of falling and less chance of occurring hyperextension. FES cycling is used to improve musculoskeletal fitness and cardiopulmonary functions (Arnold et al., 1992). The greater part of research about FES cycling use stationary tricycles and wheelchairs, but some use a tricycle as a way of transportation (Berkelmans, 2008).

The cycling movement has a specific pattern. Right and left legs keeping alternating contraction of extensors and flexors muscles synchronously. Normally, the quadriceps group provides the force for knee extension, while hamstrings group provides force for knee flexion. Some projects also employ gluteal muscles to add more power to extension (Hunt et al., 2001). The crankset fixes the trajectory, and at a constant velocity, the contraction pattern is the same for each cycle. Hence, it is possible to associate a stimulation pattern with the crank angle. The pattern may be different if the velocity changes (de Sousa et al., 2016). Also, many other factors can change the stimulation pattern, e.g., disturbances and fatigue. Figure 3.1 shows an example of a relationship between crank angle e muscle groups. The colored bars indicate when the muscles are stimulated.

Some controllers use only the crank angle to define which muscle groups are stimulated using a pre-defined on-off pattern. The stimulation phase for each muscle group is specific for a given participant. Then they may not work properly for other participants since the controller cannot correct fluctuation, and the activation level changes over time (Abdulla et al., 2014). Other alternative is to control the cadence of the crank. Conventional controllers are proportional-integral-derivative (PID) controller (Chen et al., 1994) and fuzzy logic controller (FLC) (Chen et al., 1997). PID uses as input the error between the desired velocity and the crank velocity, and the PID equation gives the output signal. All parameters of the PID equation depends on the model and are defined *a priori*. The FLC uses as input the error between the desired velocity and the crank velocity, and the change in error obtained by subtracting the last sampling velocity from the new one. The inputs are normalized by scaling factor and fuzzified. Then the algorithm infers an action according to a rule table. Finally, the stimulation level is sent to the FES cycling plant after defuzzification and scaling.

As neuromusculoskeletal models are often nonlinear and difficult to model mathematically, some studies use model-free strategies. Model-free strategies use other methods than mathematics. Hence, they do not need a detailed description of the system, which in some cases is not available. FLC is the most common (Chen et al., 1997; Abdulla et al., 2014), but researchers also use other controllers like neural network (NN) (Kim

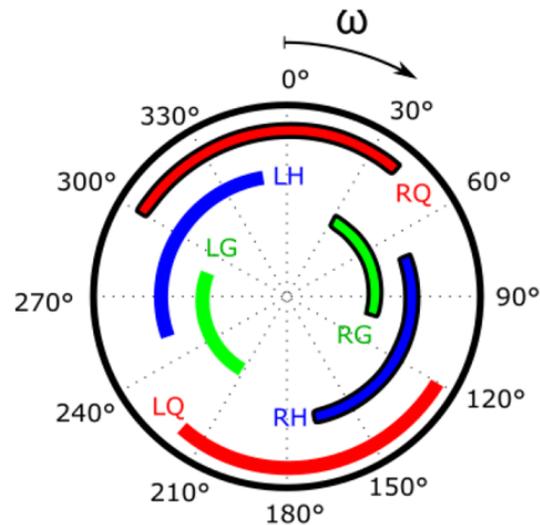


Figure 3.1: Relationship between crank angle and muscle groups activation. 0° is right leg upward, and direction is clockwise. Muscle groups are right quadriceps (RQ), right hamstrings (RH), right gluteus (RG), left quadriceps (LQ), left hamstrings (LH) and left gluteus (LG).

et al., 2008; Li et al., 2010). The controller uses a pair of mutually inhibiting neurons that work like a neural oscillator (Matsuoka, 1985). The difference between their outputs controls the extension and flexion of a joint. A sensory feedback term gives the controller characteristics similar to a first-order delay system (Kim et al., 2008; Hase et al., 2002). This feedback can be the crank angle error, velocity error, or a combination of them.

3.2 FES ROWING

FES rowing is another application with a rhythmic pattern. But different from FES cycling, it consists of a total body exercise in which FES controls knees flexion and extension to slide a seat on a track while the participant voluntarily pulls a handle. The addition of voluntary upper body exercise improves cardiovascular parameters like stroke volume and metabolic rate compared to FES leg exercise alone (Mutton et al., 1997; Davoodi et al., 2002).

Figure 3.2 shows the stages of an FES rowing exercise. First, the participant assumes the Catch position (i.e., both knees fully flexed) (Davoodi et al., 2002). Then, FES control enters the Drive state in which the knees extend until the Handle Pull position. In the Handle Pull position, the knees maintain full extension while the participant pulls the handle. After finishing the pulling, the control enters the Recovery state in which the knee flexes and goes back to the Catch position. Normally, the quadriceps group provides the force for knee extension, while hamstrings group provides force for knee flexion, but some projects have also used gluteal muscles to add more power to extension (Ektas et al., 2014).

Some of the initial works concerning FES rowing were Laskin et al. (1993); Davoodi et al. (2001). Davoodi et al. modified an indoor rowing machine for FES rowing. As safety measures, a seating system was developed to provide better trunk stability for paraplegic users, and a two-bar mechanism was attached to the seat to constrain the motion of the legs to the sagittal plane. Additionally, they installed safety stops on the rail to limit the range of the seat motion and prevent hyperextension and hyperflexion of the knee joint. Other projects employed similar safety measures (Draghici and Shefelbine, 2014; Ektas et al., 2014).

The simpler controller for FES rowing is a manual system. The participant press command buttons on the

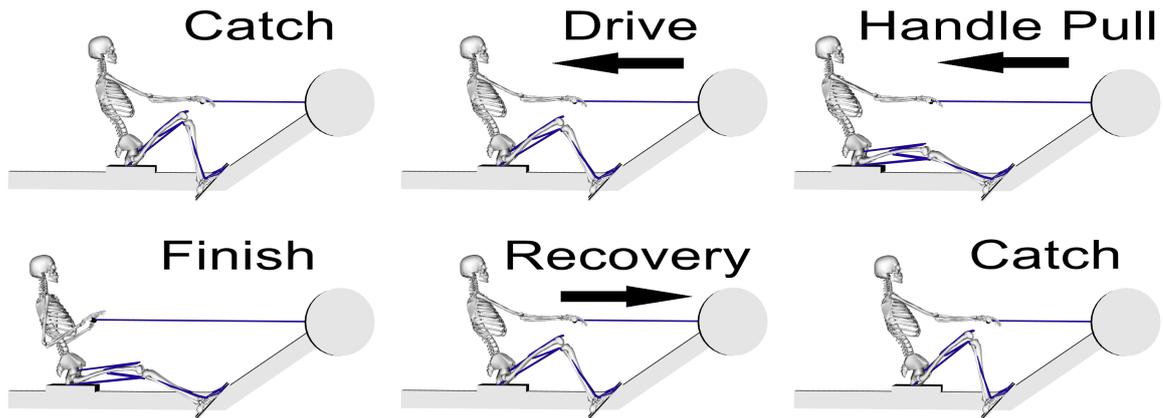


Figure 3.2: Stages of an FES rowing exercise. It starts at the top left, followed by the states in the same line, and ends at the bottom right.

handle bar to control which muscle group is stimulated by a fixed signal (Davoodi et al., 2002; Draghici and Shefelbine, 2014; Ektas et al., 2014). One button controls the knee extension, while other controls the knee flexion. During the handle pulling motion, the participant has to press the button for knee extension so the knees can withstand the arm pulling force.

The manual controller is simple and effective, but a study in which participants used an ergometer with manual and automatic control demonstrated that participants prefer the automatic control (Davoodi et al., 2002). Instead of pressing buttons manually, a finite state controller identifies the state of the movement and applies a constant stimulation to the muscles associated with that state. It is possible to define the states using sensors on the handle (Yusoff et al., 2014) or both handle and seat (Davoodi et al., 2002). As a result, the automatic controller did not require repetitive presses on the buttons and required less concentration. However, at least in Davoodi et al. (2002), the automatic controller spends more electrical stimulation per cycle than the manual controller.

In an attempt to reduce the stimulation cost of the automatic control system, a series of low-level controllers were used to replace the constant stimulation. Davoodi and Andrews (2003) have tried to use a PD controller in which the output varies according to the errors of seat position and velocity. The target position is the Handle Pull position for the Drive state and the Catch position for the Recovery state. An important characteristic of this PD controller is that a negative output stimulates the muscles which decelerate the movement. However, participants with paraplegia found it difficult to coordinate the voluntary upper body and the FES assisted lower limb movements.

A more suitable approach for the coordination problem may be the FLC. Experiments demonstrated that the stimulation level with FLC is similar to the manual controller (Davoodi and Andrews, 2004). Also, it is much easier than the manual control because the participant does not need to press buttons and can concentrate on the coordination. Davoodi and Andrews (2004) have used an FLC for the Drive state and one for the Recovery state, and a constant stimulation during the handle pulling motion. Hussain et al. (2010) have used an FLC for each state. The developed FLC used as input the position and velocity of the seat or the corresponding errors. The inputs were normalized by scaling factor and fuzzified. Then the algorithm inferred an action according to a rule table. At last, the stimulation level was sent to the stimulator after defuzzification and scaling.

3.3 SIMULATIONS STUDIES FOR FES CONTROL

Tests with subjects using a real system make the early elaboration of controllers difficult. Also, the dynamic characteristics of the human body are difficult to predict. Computer simulations are used to test a high range of strategies and parameters in a short time. Despite the fact that musculoskeletal models are limited and simpler than a human, some researchers developed model-free controllers, which are easier to port to a real system (Chen et al., 1997; Abdulla et al., 2014; Kim et al., 2008; Li et al., 2010).

Simulations are also useful to compare different strategies to see which is better and try different configurations. It is possible to change posture, load, and sensibility of the sensors adding just a small number of lines in the code.

As the environmental conditions of a computer simulation are controlled, it makes easier the use of optimization algorithms to find optimal parameters for each controller. Previous studies have used genetic algorithm (GA) (Abdulla and Tokhi, 2014; Kim et al., 2008; Yahaya et al., 2014; Hussain et al., 2010) to improve performance, and Yahaya et al. (2014) also have used particle swarm optimization (PSO). FLC and NN have feature many parameters, which justify the use of optimization algorithms.

GA is not so good to find the globally optimized solution, but it may find a good result in a reasonable number of iterations (Kim et al., 2008). For FES applications, the globally optimized solution is not necessarily important because many factors reduce the precision of simulations, such as the difference between the human body and the musculoskeletal model. Then, control design must be focused on robustness instead of precision.

In terms of simulation studies, models can be simple 2D bar linkage models (Chen et al., 1997; Gföhler and Lugner, 2004; Kim et al., 2008) to a complex 3D musculoskeletal model (Abdulla and Tokhi, 2014; Li et al., 2010; Yahaya et al., 2014; Hussain et al., 2008). In the case of 3D models, the projects that we know use Matlab or Vn4D with controllers developed in Matlab. These programs have tools for analysis and visualization of data, which facilitates the comparison of results. Each project has its musculoskeletal model. Hence, compare results between them has its limits because the model and simulation variables are different for each model. OpenSim has predefined muscles, joints and types of contact, which facilitates the development of new applications and comparison of results. However, as far as we know, there is no work where OpenSim was applied to investigate FES cycling and FES rowing.

4

BASIC FRAMEWORK

This chapter presents the basic framework to simulate musculoskeletal models particularly in FES applications that feature a rhythmic pattern. First, the development of musculoskeletal models is presented. We used a standard model from OpenSim and added new objects and properties using Blender and Matlab. Matlab scripts available by the OpenSim community allowed us to implement closed-loop control strategies. As the main objective of an FES application is contracting muscles to generate the desired motion, we created a general framework for dynamic simulations that is presented in the next section. In particular, we focused in rhythmic movements, such as cycling and rowing. We also added some environment changes to analyze the behavior of controllers, because real environments have many factors that change the results, like noise, load, and fatigue. Additionally, we took advantage of the capacity of running a high number of simulations in a short period to optimize the solutions using GA. At last, this chapter presents a method to compare the efficiency of different FES applications using OpenSim tools.

4.1 MUSCULOSKELETAL MODEL

OpenSim includes default musculoskeletal models to create basic applications. These musculoskeletal models were based on (Delp et al., 1990)¹, representing a healthy adult subject with a height of 1.8 m and mass of 75 kg. There are models with a different number of degrees of freedom and muscle-tendon actuators to use in specific applications, e.g., a primarily lower extremity model can have 23 degrees of freedom and 92 muscle-tendon actuators or only 10 degrees of freedom and 18 muscle-tendon actuators. The later is for fast prototyping, as the time of simulations increases when the model is more detailed.

Members of OpenSim community also distribute their musculoskeletal models, as the modified and expanded version of the lower extremity model, which contains 29 degrees of freedom and 92 muscle-tendon actuators, including arms with inertial properties, but no muscles (Hamner et al., 2010). This modified model is adequate for applications like FES rowing, which needs feedback from the arms, but no actuation.

We modified some available musculoskeletal models to use in FES applications, adding new bodies (solid objects), joints, muscles and forces. The goal is to recreate the real applications. Hence, the models needed the equipment of each exercise or at least part of them. We used Blender to create new bodies (Blender, Online Community, 2013). Blender is open source and can export to the basic format .obj, an open data format to represent simple 3D objects. OpenSim can read .obj files and use them as bodies. Figure 4.1 shows cycling and rowing equipment made using Blender.

We used measures from real pieces of equipment to create the bodies. As some parts of the equipment have degrees of freedom, these parts had to be in different files. The new objects were added using the OpenSim package for Matlab² and linked to each other and the musculoskeletal model using joints. Joints, muscles and constraints are inserted in a body or connect two bodies in a specific position and orientation. Hence, as the bodies, they need to use measures from the real pieces of equipment.

Orientation needs special attention when converting objects to OpenSim. Figure 4.1.b shows a rowing machine in Blender using the front view. The X-axis is pointing to the right direction, Y-axis forwards and Z-axis upwards. In OpenSim, X-axis is pointing to the right direction, Y-axis upwards and Z-axis backwards. Hence, while adding new joints, we had to convert translations and rotations according to the OpenSim standard.

¹It is being developed in maintained in <https://simtk.org/projects/opensim>

²<http://simtk-confluence.stanford.edu:8080/display/OpenSim/Scripting+with+Matlab>

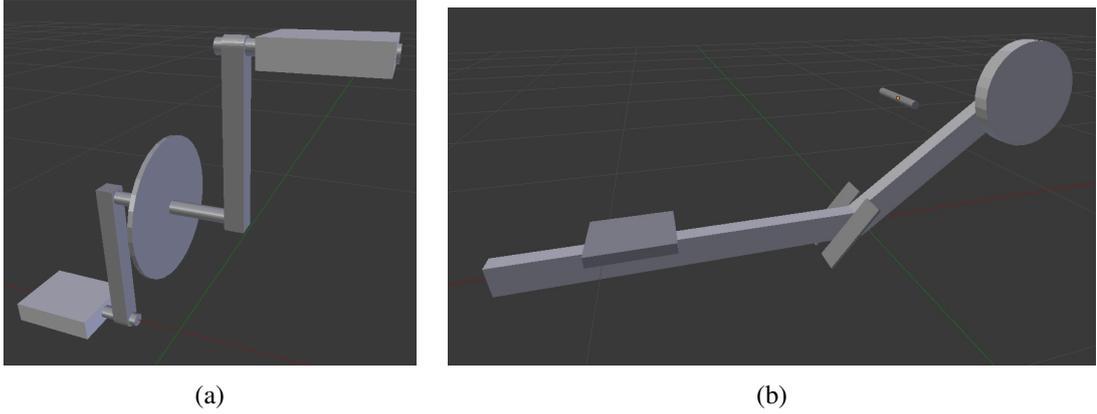


Figure 4.1: (a) Cycling and (b) rowing models designed using Blender. Dimensions are based on measurements from real equipment.

4.2 FRAMEWORK FOR CLOSED-LOOP CONTROL SIMULATIONS

Even though FES applications have their specific characteristics, they have the same objective, which is contract muscles to generate the desired motion. In a closed-loop system, the controller tries to minimize the difference between the desired motion and the actual motion the model performs. Then, the output of the controller activates the muscles of the model. In OpenSim, the muscle activation has the range $[0, 1]^3$. If the value is 0, the muscle is relaxed, and if the value is 1, it applies the maximum isometric force.

Some Matlab scripts of the OpenSim package⁴ execute forward dynamics simulation step by step. Normally, OpenSim executes an open-loop forward dynamics. It is possible to activate muscles and read parameters in each step with the Matlab scripts, closing the loop of the system. Besides, these Matlab scripts can use Matlab ordinary differential equations (ODE) solvers⁵. In this work, we used the same sampling time of 0.02 s for the control frequency that was used in a previous work (de Sousa et al., 2016).

One of the characteristics that make a rhythmic motion a complex problem is the different actuation of the muscles. Each muscle has its activation profile, but they contract in a coordinated way to execute the movement as a whole. Segmenting the rhythmic motion into states and choosing which muscles will be activated may be a better strategy than using one controller per actuator. Hence, a higher level finite state controller (FSC) was inserted in the control strategy to decide which muscles will be activated by the lower level controller. The FSC is of particular importance for FES rowing because during the handle pull phase the controller has to apply a constant stimulation to keep the knees fully extended while the participant pulls the handle (Davoodi et al., 2002). Note that it is possible to use the FSC with a constant activation to perform an on-off pattern control, as the FSC will choose the muscles that belong to the motion states.

The basic framework of the control strategy is illustrated in Figure 4.2. The Musculoskeletal Dynamics block has as input a vector (U) with the activation of all muscles used in the strategy. We used two types of inputs (I): A step input that change the desired velocity without transition; and a ramp input that does a ramp transition through the equation

$$\omega = (\omega_2 - \omega_1)(t - t_{r1}) / (t_{r2} - t_{r1}) + \omega_1, \quad (4.1)$$

where ω_1 is the old velocity, ω_2 the new velocity, t_{r1} the time to begin the ramp and t_{r2} the time to end it.

³https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1Muscle.html

⁴<http://simtk-confluence.stanford.edu:8080/display/OpenSim/Scripting+with+Matlab>

⁵<https://www.mathworks.com/help/matlab/math/choose-an-ode-solver.html#bu3n41b>

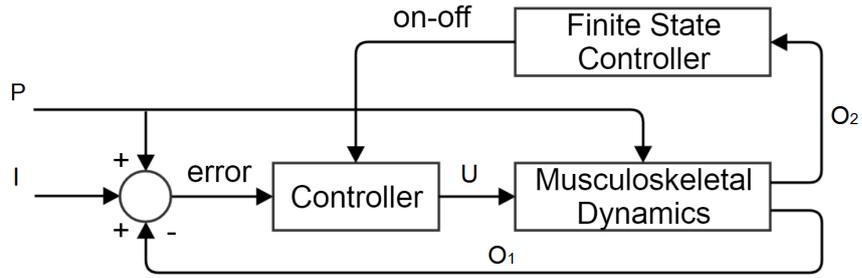


Figure 4.2: Basic framework. The Musculoskeletal Dynamics block represents the simulation OpenSim performs with the Matlab scripts. The Controller block is the lower level controller that changes the amplitude of activation (U) according to the error between the input (I) and (O_1). The Finite State Controller block chooses which muscles will be activated by the lower level controller according to (O_2) to find the actual state of the dynamics. The input vector P changes the environment, i.e., adding noise, load, and fatigue.

4.3 MODIFYING THE ENVIRONMENT

The musculoskeletal model is quite simple compared to a person, e.g., it does not consider the impedance of the tissue and recruitment of unintended muscles due to diffusion of electrical stimuli (Popović and Sinkjær, 2012). Additionally, besides the smaller number of degrees of freedom and muscles, OpenSim has a limited physics. For example, there is no friction between bodies linked by a joint, and the system does not contain imperfect sensors. Then, we added some properties to simulate real conditions.

4.3.1 Fatigue

One of the main reasons we used simulations is the accelerated muscle fatigue caused by the functional electrical stimulation during real tests (Popović and Sinkjær, 2012). Fatigue reduces the time for tests. Then it is quite difficult to test a large range of parameters for each controller. However, the OpenSim standard muscle-tendon model does not simulate fatigue. We had to add fatigue to the model, i.e., we changed the relationship between the control signal and muscle activation. We used the first order equation

$$\dot{f} = \frac{1}{T_f}(1 - f)u + \frac{1}{T_r}(-f), \quad (4.2)$$

where f is the fatigue factor, \dot{f} the derivative of f , u the control signal, T_f the fatigue time constant and T_r the resting time constant. The new control signal is

$$u = (1 - f)u. \quad (4.3)$$

Fatigue increases when there is muscle excitation (u), but saturates in 1. It exponentially decays until 0 when there is no stimulation. Note that the limit borders are smooth, i.e., there is no discontinuity. The constants T_f and T_r control the velocity in which the fatigue increases or decreases. FES sections with participants with SCI last a few minutes to an hour (Hunt et al., 2002), while computer simulations last a few seconds (Abdulla et al., 2014). Hence, time constants have to be small to see the fatigue effect during simulations. Figure 4.3 shows an example of muscle activation with fatigue which $T_f = 30$ s and $T_r = 30$ s.

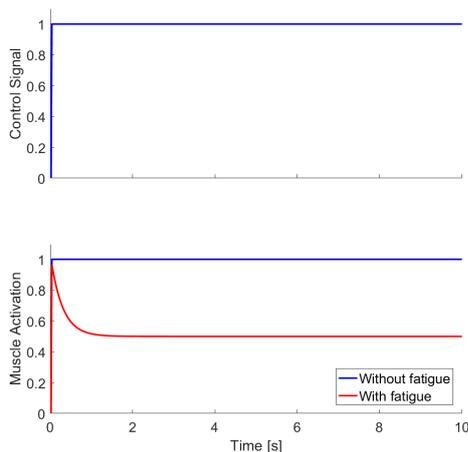


Figure 4.3: Control signal (top graph) and muscle activation (bottom graph) without (blue line) and with (red line) fatigue. The maximum muscle activation is equivalent to 1. The muscle is resting when the activation is 0. The time constants of the fatigue equation are $T_f = 30$ s and $T_r = 30$ s.

4.3.2 Other modifications

Other modification was feedback with noise to simulate imperfect sensors. We added a white noise to positions, rotations, and velocities used to calculate the error and the next state of the FSC. Figure 4.4.b shows an example of angular velocity with white noise which maximum absolute value is 5° .

For FES cycling, we added a load to the crank to increase the effort needed. The real adapted tadpole trike (HP3, Brazil) from which the dimensions to create the musculoskeletal model were taken has been used in combination with a weight (see Figure 4.5). First, we decided to implement a simple torque against the movement of the crank (a more detailed explanation is provided Chapter 5). The torque is multiplied by the angular velocity of the crank so that the torque is null when the crank is not moving. However, this method increases the torque proportionally to velocity. Also, OpenSim does not report the force applied to the crank.

We also tried to use a damping force⁶ to increase the effort. It changes the effort needed according to the damping coefficient, and OpenSim can calculate the crank torque using the Analyze Tool. However, the damping force does not have a maximum value, and it increases proportionally to the deviation of two frames. We decided to use it because we can record the force applied to the crank.

4.4 GENETIC ALGORITHM

We decided to use GA to find the parameters of some controllers for FES applications. The set of control parameters is represented as an individual of the GA. Then, a set of individuals (population) go through the stages of reproduction, crossover, and mutation (Coley, 1999; Mitchell, 1999). At last, the best individuals evaluated by a fitness function are selected for the next generation.

The objective in using GA was not to find the optimal solution according to the fitness function, but a solution good enough to perform a stable motion even with noise, load, and fatigue. An optimal solution could be too specialized, i.e., it could only work for a specific musculoskeletal model and environment configuration. Besides that, the time cost of simulations restricted the number of times GA could be used. For example, a GA

⁶https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1BushingForce.html#details

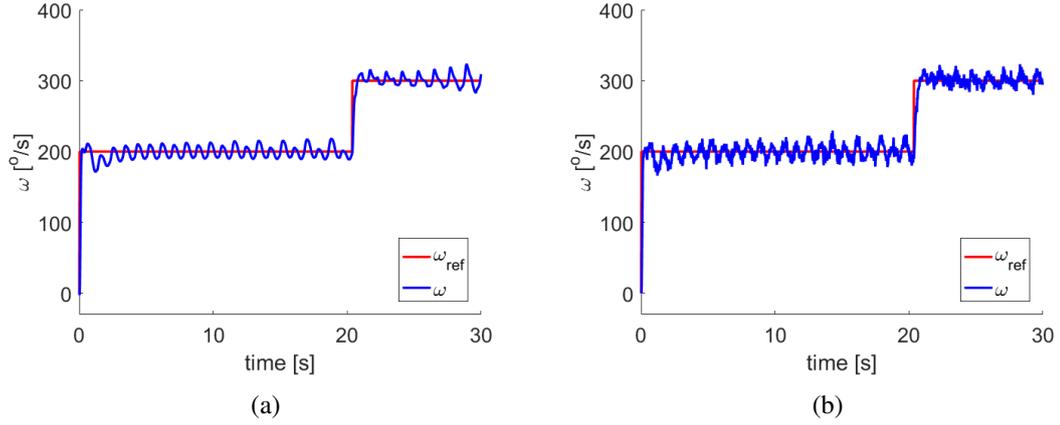


Figure 4.4: Angular velocity (a) without and (b) with white noise which maximum absolute value is 5° . The red line is the desired velocity, and the blue line is the actual velocity. The modification is added to the intrinsic oscillation of the system.

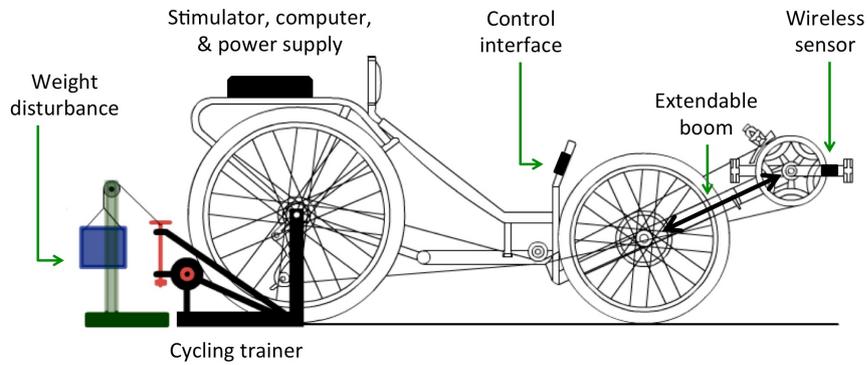


Figure 4.5: Components of the adapted tadpole trike (HP3, Brazil). On the bottom left, a weight system to change the load (da Fonseca et al., 2016).

with a population size of 10 and a maximum number of generations of 10 takes approximately one day to finish using the lower limb model with only 10 degrees of freedom and 18 muscle-tendon actuators. A GA with the same configuration takes about four days to finish using the modified and expanded version of the lower limb model (Hamner et al., 2010).

To avoid a specialized solution, the GA evaluated only results of simulations with step input and without environment modifications. Furthermore, the parameters obtained using GA were reused to create controllers which are part or combination of others controllers. We used a small population size of 10 and a maximum number of generations of 10. The probability of crossover was 0.8, and the probability of mutation was 0.15. We used a high probability of mutation because the population size and the maximum number of epochs were small. A high probability increases the number of tested values, which increases the possibility of finding a better solution. We used elitism to avoid loose an existing good solution during crossover and mutation. The selection of the next generation was made using the roulette wheel selection method (see Algorithm 4.1).

The interval $[0, 1]$ represents the area of the wheel. It is split into N sections, where N is the population size. The size of the k th section will depend on the cumulative probability of selection of the k th individual, i.e., the sum of the probability of all previous individuals until the k th individuals. The limits of a section k are the cumulative probabilities of the $(k - 1)$ th and k th individuals. The probability depends on the normalized fitness function. It is divided by the total sum of the normalized fitness values, hence the sum of all probabilities (i.e., the cumulative probability of the Nth individual) will be 1. The roulette method chooses a random number

Algorithm 4.1 Roulette method for selection of the next generation.

- 1 Calculate the *fitness* function for the individuals: $f(x^k)$
 - 2 Calculate the normalized *fitness* function for the individuals: $f_n(x^k) = \frac{f(x^k) - \min(f(x^k))}{\max(f(x^k)) - \min(f(x^k))}$
 - 3 Find the total normalized *fitness* of the population: $S = \sum_{k=1}^N f_n(x^k)$
 - 4 Calculate the probability of selection of the individuals: $p^k = \frac{f_n(x^k)}{S}$
 - 5 Calculate the cumulative probability: $q^k = \sum_{j=1}^k p^j$
 - 6 **for** $i = 1 : N$
 - 6.1 Generate a random number r_{nd} in the interval $[0, 1]$
 - 6.2 If $r_{nd} < q^1$, choose x^1 . If $q^{k-1} < r_{nd} \leq q^k$, choose x^k
 - 6.3 **endfor**
-

in the interval $[0, 1]$. The higher the fitness value, the greater the chance of an individual being chosen.

The fitness function will depend on the problem, but in most of the cases, the problem is to minimize the position error or velocity error. We can add other measures to avoid, for example, “asynchronous” movements between the legs.

4.5 MUSCLE WORK RATE AND ENERGY COST EFFICIENCY

Hunt et al. (2012) recommended the total work efficiency and the stimulation cost as benchmark measures. It was first suggested to estimate the performance of different control strategies for FES cycling. We decided to use this benchmark in order to compare FES applications since both cycling and rowing produce mechanical work and the movements have a metabolic energy cost. However, this benchmark is for real tests, and it estimates the metabolic energy cost of steady-state aerobic work from the rates of pulmonary oxygen uptake and carbon dioxide output.

The metabolic energy cost we used was calculated using an OpenSim tool based on (Umberger et al., 2003). The total rate of the proposed muscle energy expenditure model, which is expressed in W/kg of muscle tissue, is the sum of the thermal and the mechanical energies liberated during a steady-state motion. The general equation for the full activation of a muscle is

$$\dot{E} = \dot{h}_{AM} + \dot{h}_{SL} + \dot{w}, \quad (4.4)$$

where \dot{h}_{AM} is the activation and maintenance heat rate, \dot{h}_{SL} the shortening/lengthening heat rate, and \dot{w} the mass specific mechanical work rate.

The proposed model was based mostly on mammalian muscle data (Umberger et al., 2003). The main feature of this model is the energy expenditure during submaximal activations. The energy is scaled so that \dot{h}_{AM} is increased and \dot{h}_{SL} is reduced compared with a linear scaling. \dot{h}_{AM} increases at low activation due in part to increased energy variation associated with force and length oscillations, and the muscle has a rapid rise and slow decay of heat production at the beginning and end of activation, respectively.

We calculated the total metabolic energy cost as the sum of the energy expenditure of the muscles used by the FSC since the musculoskeletal models used for FES cycling and FES rowing have different numbers of muscles. As to compare the performance of FES cycling and FES rowing, the energy cost efficiency was defined as the average total work over the average total metabolic energy cost:

$$\eta = \frac{\bar{w}}{\bar{E}} \times 100\%. \quad (4.5)$$

5

FES CYCLING

This chapter presents the development of a musculoskeletal model and control strategies for FES cycling. FES cycling is simpler and safer than standing and walking. The cycling movement has a specific pattern, which is right and left legs keeping alternating contraction of extensors and flexors muscles in a rhythmic pattern.

First, the musculoskeletal model is presented. We used a standard model from OpenSim and added new objects and properties using Blender and Matlab. We used the measures of the adapted tadpole trike (HP3, Brazil) (see Figure 5.1) to develop the additional parts of the model. The chapter also contains the description of coupled oscillators adapted for FES cycling. Then, the next section presents a series of simulations involving different control strategies and modifications in the environment. The objectives of these studies are comparing which control strategy is more effective with most of environment modifications and improve controllers for the real system. Also, the adapted tadpole trike can be used for different participants, who have different heights and weights. Hence, we performed simulations with scaled models. The chapter concludes presenting the results of energy expenditure for FES cycling.

5.1 MODEL

5.1.1 Initial model

We used the simplified model of lower limbs included in OpenSim (Delp et al., 2007) as a starting point for the FES cycling model. It was chosen because its simplicity allows fast simulations. It features head-trunk, pelvis and legs segments.

The muscles in the model are Hamstrings (*hamstrings*), Biceps Femoris Short Head (*bifemsh*), Gluteus (*glut_max*), Iliopsoas (*iliopsoas*), Rectus Femoris (*rect_fem*), Vastus Lateralis (*vasti*), Gastrocnemius (*gastroc*), Soleus (*soleus*) and Tibialis Anterior (*tib_ant*). The quadriceps group consists of Rectus Femoris and Vastus Lateralis.

The degrees of freedom of the model include the movement of the pelvis (*pelvis_tilt*, *pelvis_tx*, *pelvis_ty*), hip (*hip_flexion*), knee (*knee_angle*), ankle (*ankle_angle*) and lumbar (*lumbar_extension*). The names in paren-



Figure 5.1: The adapted tadpole trike (HP3, Brazil). We used the measures of this tricycle to develop the additional parts of the FES cycling model (Bó et al., 2017).

thesis are the equivalent objects in the musculoskeletal model file.

5.1.2 Added objects

To simulate the cycling movement, we added five objects to the model: two foot supports, two pedals, and one crankset. The crankset is fixed on the ground, allowing only rotation along the sagittal plane. The pedals are attached at the end of the crank arms and can rotate along the axis perpendicular to the crank arms.

The foot supports were used to fix the feet on the pedals. Initially, they were supposed to be attached to the pedals using physical contact. The pedals were placed on the foot supports, which had a box shape, and the property of contact geometry¹ allowed the pedals to collide with the box. However, this increased the number of calculations, increasing computation time by 150%, and the collision physics had limitations (e.g., in some occasions, one body pass through the surface of the other body).

As a new solution, we added the new objects to the kinematic chain of the model. The simplest method is to add the new objects to another kinematic chain, making the crankset a child of the ground, and the foot a child of the foot support. However, this method creates the problem of a loop in the kinematics, i.e., the foot belongs to two kinematic chains. OpenSim avoids this problem using constraints². Constraints are objects that limit motion between two connected bodies. They are similar to joints, but it is not possible to add an actuator and then control constraints. In the case of the *PointOnLineConstraint*, it only allows the rotation along a line. It is important to note that a constraint automatically adjusts the angles and translations of the joints when the state of the model changes, which is similar to what inverse kinematics does. Constraints have a narrow margin of tolerance, and the forward dynamics will stop if the tolerance is not satisfied.

The new objects were added as follows:

- Foot support as a child of foot (*calcn*) with *WeldJoint*;
- Pedal as a child of Foot support with *WeldJoint*;
- Crankset as a child of ground with *PinJoint*;
- Pedal connected to Crankset with *PointOnLineConstraint* perpendicular to the crank arm.

Note that the pedal could be a child of a foot. In this version of the model, the foot supports are used only for aesthetic purpose.

5.1.3 Cycling pose

The FES cycling model was used to create new controllers for the real system. Hence, the pose of the model had to resemble a person using the equipment (see Figure 5.1). Especially, we had to lock the ankle at a right angle to simulate the foot orthoses used in the real system. Also, others degrees of freedom were locked to represent the sitting pose and bike geometry. The initial angles and which joints are locked are shown in Table 5.1. Figure 5.2 shows the musculoskeletal model in the cycling pose. It is possible to simulate other bikes changing the parameters of Table 5.1.

¹https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1ContactGeometry.html

²<https://simtk-confluence.stanford.edu/display/OpenSim/OpenSim+Models#OpenSimModels-KinematicConstraintsinOpenSim>

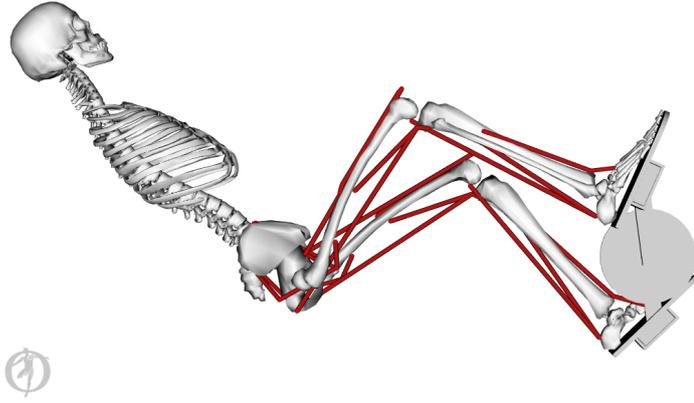


Figure 5.2: Cycling pose. The pose of the model had to be similar to the pose in the real system (see Figure 4.5). Head-trunk and pelvis were fixed to simulate a person sitting on the tricycle.

Table 5.1: Initial angles and which joints are locked in the FES cycling model. The joint names follow the nomenclature from OpenSim standard.

Joint	Initial Angle ($^{\circ}$)	Locked
pelvis_tilt	44.712	yes
pelvis_tx	-0.708	yes
pelvis_ty	0.052	yes
hip_flexion_r	104.036	no
knee_angle_r	-82.716	no
ankle_angle_r	0.000	yes
hip_flexion_l	77.291	no
knee_angle_l	-78.207	no
ankle_angle_l	0.000	yes
lumbar_extension	0.000	yes
crankSetToGround	7.5113	no

5.1.4 Load

One limitation of using a purely musculoskeletal model to test new controllers is the lack of a load. In the real system, the weight of the bicycle, the weight of the pilot and the friction change the amount of necessary force to generate motion.

To simulate this load, we added to the model a force opposing the force applied on the crank. In OpenSim, actuators generate force. We used a *TorqueActuator*³ attached to the ground and the crank along the opposite axis in which the crankset rotates.

Additionally, we added a simple controller to the simulated load. When the torque is constant, and there is no actuation of the muscles, the model would rotate backward. Also, there is a high muscle activation during the beginning of the simulation to overcome the constant load. To avoid this, the torque is multiplied by the angular velocity of the crank so that the torque is null when the crank is not moving. Note that this method does not have a maximum limit, and torque increases proportionally to velocity.

We also tried to use a *BushingForce*⁴ to increase the effort. It was used in the study to evaluate the performance of the controllers with different levels of load. *BushingForce* changes the effort needed according to

³https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1TorqueActuator.html

⁴https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1BushingForce.html#details

the damping coefficient, and OpenSim can record the force applied in the crank, so it is possible to record the crank load. This method also does not have a maximum limit.

5.2 COUPLED OSCILLATORS FOR FES CYCLING

Participants of FES exercises have different physical characteristics, and these characteristics also change through time. Besides that, muscle contraction is not linear (Umberger et al., 2003). Hence, nonuniform oscillators are better suited to couple the controller and model for FES applications than constant oscillators. Additionally, a controller which can change its frequency is necessary to correct the frequency of the model. The two differential equations for the controller and model are

$$\dot{\theta}_{c1} = \omega_{c1} + a_{c1} \sin(\theta_m - \theta_{c1}), \quad (5.1)$$

$$\dot{\theta}_m = \omega_m + a_m \sin(\theta_{c1} - \theta_m), \quad (5.2)$$

where $\omega_{c1}, \omega_m > 0$ rad/s and $a_{c1}, a_m > 0$ rad/s. $c1$ represents the controller with a nonuniform oscillator, and m represents the model. The parameters are chosen considering that

$$|\omega_{c1} - \omega_m| \leq a_{c1} + a_m. \quad (5.3)$$

θ_m is calculated through sensors in the equipment/simulation, which is used to calculate $\dot{\theta}_{c1}$. The angle θ_{c1} is then calculated as

$$\theta_{c1} = \dot{\theta}_{c1} \Delta t + \theta_{c1}, \quad (5.4)$$

where Δt is the time between two samples of the simulation. θ_{c1} is sent to a high-level controller in order to choose the muscles to be excited. Also, note that the amplitude of the circle is constant. Hence, coupled oscillators do not calculate the intensity of the activation. $\dot{\theta}_{c1}$ is used as an input for a PI controller to control the activation of the muscles. The PI controller uses $\dot{\theta}_{c1}$ to calculate the error and to generate the activation. The activation u is calculated as

$$e = \omega_{ref} - \dot{\theta}_{c1}, \quad (5.5)$$

$$e_i = e_i + e \cdot t, \quad (5.6)$$

$$u = K_p \cdot e + K_i \cdot e_i, \quad (5.7)$$

where ω_{ref} is the desired frequency.

Both oscillators converge to the compromise frequency ω^* , and ω^* depends on not only the parameters of the controller but also the parameters of model. That means the coupled oscillators can adapt the input of the PI controller based on the model characteristics. However, the compromise frequency is not necessarily the desired frequency. For the FES application controllers which minimize the error to achieve the desired frequency, it is necessary another oscillator with a constant frequency equal to ω_{ref} . The angle of this third oscillator is then

$$\theta_{c2} = \omega_{ref} \Delta t + \theta_{c2}, \quad (5.8)$$

where $c2$ represents the controller with a constant oscillator, and Δt is the time interval. A system diagram with the coupled oscillators adapting the input is shown in Figure 5.3.

Despite the fact that the oscillator has a constant frequency, the desired frequency ω_{ref} is controlled by the user. The addition of this oscillator ensures that the three oscillators will converge to ω_{ref} . The controller with nonuniform oscillator was modified to couple with both model and constant oscillator so $\dot{\theta}_{c1}$ will be constant when $\theta_{c2} - \theta_m$ stops changing, i.e., when the model is moving at ω_{ref} . The new differential equations is

$$\dot{\theta}_{c1} = \omega_{ref} + a_{c1} \sin(\theta_{c2} - \theta_m). \quad (5.9)$$

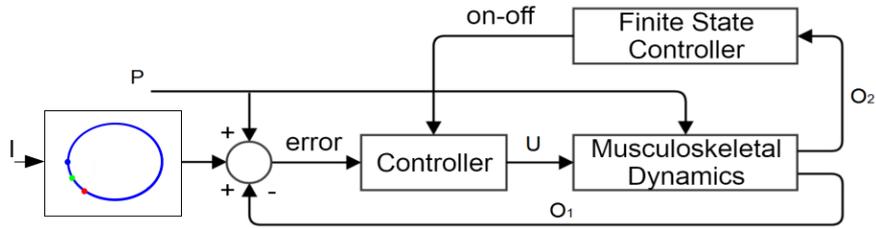


Figure 5.3: System diagram for FES cycling with coupled oscillators. The three colored points in a circular trajectory represent the oscillators m , $c1$, and $c2$. The output of $c1$ is the new input of the system.

An example of coupling of three oscillators is shown in Figures 5.4, 5.5 and 5.6. The system dynamics can be analyzed using its phase difference and frequency

$$\phi = \theta_1 - \theta_2, \quad (5.10)$$

$$\dot{\phi} = \dot{\theta}_1 - \dot{\theta}_2 = \omega_1 - \omega_2 - (a_1 + a_2) \sin(\theta_1 - \theta_2), \quad (5.11)$$

for the three possible combinations of oscillators. The values of the parameters for the example are $\omega_m = 2.0$ rad/s, $a_m = 2.0$, $\omega_{c1} = \omega_{c2}$, $a_{c1} = 2.0$, and $\omega_{c2} = 3.0$ rad/s.

Figure 5.4 shows the relation between $\dot{\phi}$ and ϕ . The three cases have attractor point and repulsor point. Hence, the system is stable. The coupling between $c1$ and $c2$ is always stable since they have the same natural frequency and the phase difference tends to 0 rad. The coupling with the model (m) will not be stable if the difference between its frequency and the other oscillator is too high.

It is possible to observe in the phase difference shown in Figure 5.6 that there is no phase difference between $c2$ and m since $\omega_{c1} = \omega_{ref}$ and the angle θ_{c1} is calculated based on the phase difference $(\theta_{c2} - \theta_m)$. Hence, $c1$ tries to couple with $c2$, canceling the sinusoidal term of its equation. The cancellation occurs when $\theta_{c2} = \theta_m$.

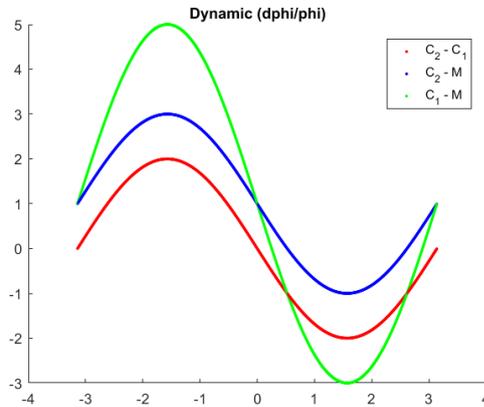


Figure 5.4: Relation between the angle (ϕ) and frequency ($\dot{\phi}$) of the dynamics of three pairs of coupled oscillators.

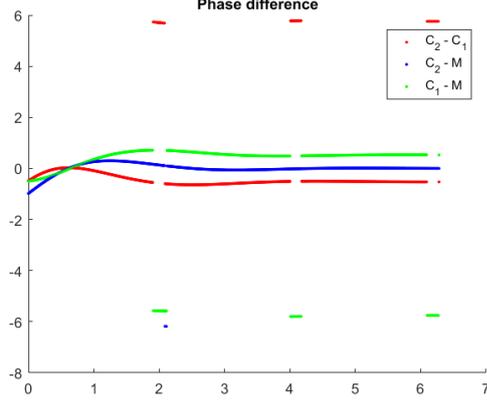


Figure 5.6: Phase difference (ϕ) of the dynamics of three pairs of coupled oscillator through time. $(\theta_{c2} - \theta_m)$ becomes 0 rad since $\omega_{c1} = \omega_{ref}$ and the angle θ_{c1} is calculated based on the phase difference $(\theta_{c2} - \theta_m)$.

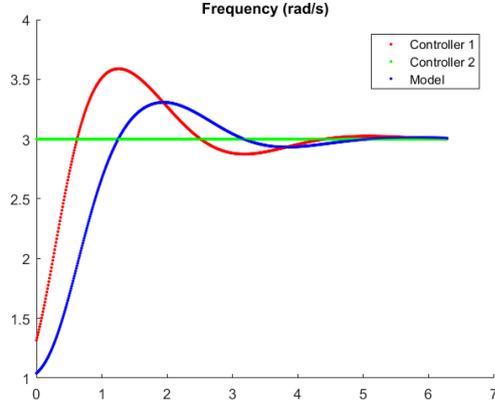


Figure 5.5: Frequency ($\dot{\phi}$) of the dynamics of three pairs of coupled oscillator through time. The frequencies of the model and the controller with nonuniform oscillator converge to the frequency of the controller with uniform oscillator.

5.3 NUMERICAL INVESTIGATION

The next subsections present the results acquired from FES cycling simulations. They were divided according to the type of highlighted feature or type of control strategy. We used the same prerequisites to acquire the data.

All simulations feature a 30s duration and sampling time of 0.02s. During the first 20s, the desired velocity is 200°/s. In the next 10 s, the desired velocity is 300°/s. The transition can be a step or a ramp. An example of simulation result is shown in Figure 5.7. If the activation of a muscle is 0, the muscle is relaxed, and if the value is 1, it applies the maximum isometric force. The “reference angle” is important to observe the coupling ability of oscillators. Control strategies used are described in the following subsections.

Additionally, we used the simulation data to calculate rise time, overshoot, maximum error in steady state, root mean square error (RMSE) in steady state and average muscle activation. We considered the steady state for the desired velocity of 200°/s (t_1) as the interval between 14 and 19 seconds. The steady state for the

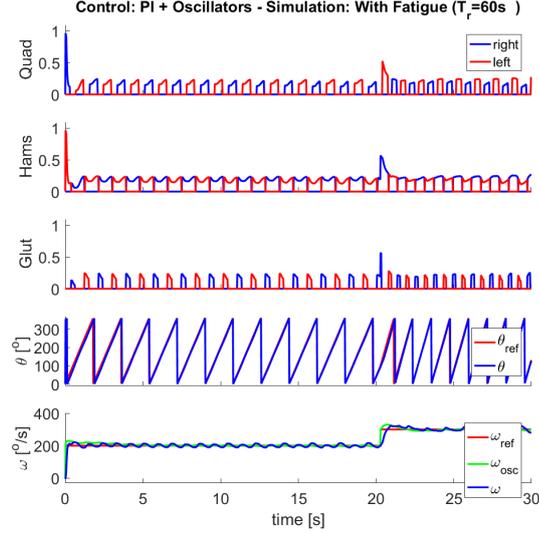


Figure 5.7: Example of a simulation result. The top three graphs are the signal sent to each muscle group, where blue lines are the signals to the right leg, and red lines are the signals to the left leg. The fourth graph is the angle ($^{\circ}$) of the crank. 0° is right leg upward, and direction is clockwise. The blue line represents the angle acquired from OpenSim, while the red line represents the angle if the velocity was always the reference velocity. The fifth graph illustrates the angular velocity ($^{\circ}/s$) of the crank. The blue line is the angular velocity acquired from OpenSim, the red line is the desired velocity, and the green line is the output of the coupled oscillators.

desired velocity of $300^{\circ}/s$ (t_2) is the interval between 24 and 29 seconds. As the performance during steady state was equal to both ramp and step input, and rise time and overshoot are calculated for the step input, we did not use simulations with ramp input to compare performance. The performance measures are the same used in a previous work (de Sousa et al., 2016). Maximum error and RMSE were calculated for both steady states. There are two approaches to find the average muscle activation. The first is to use the mean equation

$$\bar{\gamma} = \frac{\sum_{i=1}^N \gamma[i]}{N}, \quad (5.12)$$

where $\bar{\gamma}$ is the mean, $\gamma[i]$ the muscle group activation at sample i and N the total number of samples. However, this approach takes into account the samples when the activation is 0, which decreases the mean. Another alternative is to divide the sum of activations only per the number of samples with nonzero activation. For example, for a constant quadriceps activation of 0.50 during a quarter of a cycle, the average for nonzero activation is 0.50, while the total average is 0.13 for one leg and 0.25 for the two legs. The average for nonzero activation is equal to the constant activation because it contemplates only a quarter of samples.

The rise time is the time the model takes to increase the velocity from 10% to 90% of $200^{\circ}/s$, and the overshoot is the maximum peak minus the desired velocity $200^{\circ}/s$. An example of a bar chart of a performance measure is shown in Figure 5.7.

5.3.1 Tuning control parameters manually and with genetic algorithm

5.3.1.1 Previous work

In a previous work, we already did a PI controller for FES cycling (de Sousa et al., 2016). The PI controller used as input the error between the desired velocity and the crank velocity. Then, the control signal is sent to

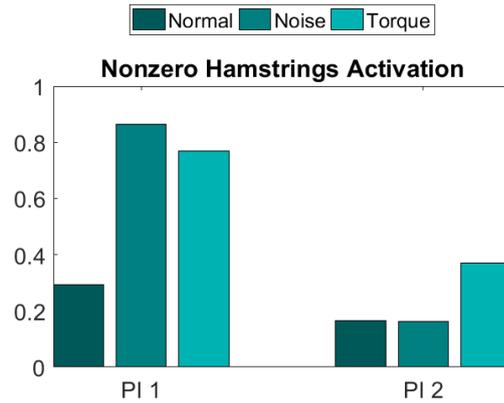


Figure 5.8: Example of a bar chart of a performance measure. The upper legend has the type of environment modification of the colors, and the groups of three bars are the performance of the control strategies.

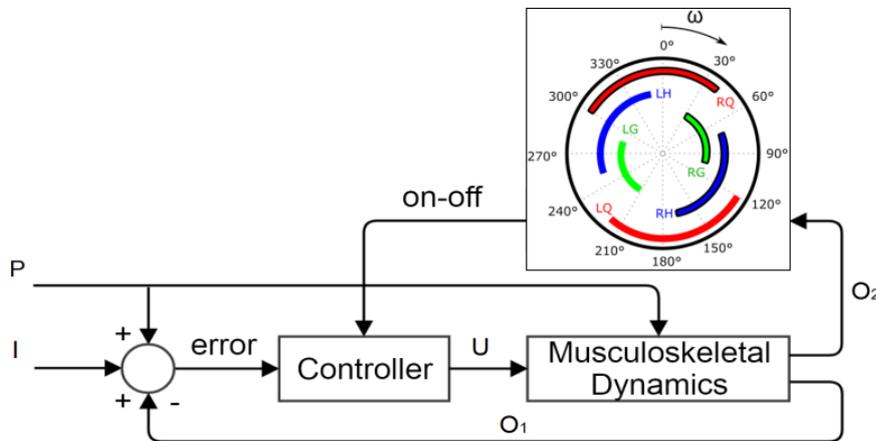


Figure 5.9: System diagram for FES cycling. The colored angle intervals of the FSC associate the crank angle with the activation of the muscles.

the muscle groups chosen by the FSC. The FSC associates the coordination of the muscles with the crank angle (see Figure 5.9). The activation profile of the muscle groups of the right leg are defined as an angle intervals in which the muscles will be activated by the control signal. The muscle groups of the left leg have an angle interval equal to the angle interval of the right leg delayed by 180° .

The constants of the PI controller and the relationship between crank angle and muscle groups activation were found empirically. The values are in Table 5.2. Angle intervals for muscle groups activation were chosen to move the legs in the clockwise direction. It is important to mention that the controller is disabled when the velocity keeps higher than the desired velocity, i.e., it is disabled when the PI output is negative. Simulations were done again with the latest version of the model and with environment modifications. We first simulated the case in which quadriceps and hamstrings are activated.

5.3.1.2 Method

We used GA to find better parameters and compared its performance with the PI controller manually tuned. The GA parameters were in accordance to the framework described in Section 4.4, and the fitness function was the sum of the absolute value of the error between the desired velocity and the crank velocity for the entire simulation. The new values are in Table 5.2.

Table 5.2: The parameters for the controllers which used quadriceps and hamstrings muscle groups. K_p and K_i are the proportional and integrative constants of the PI control. The angle interval is the crank angles when a muscle is activated. RQ is right quadriceps, and RH is right hamstrings. The muscle groups of the left leg have an angle interval equal to the angle interval of the right leg delayed by 180° . The gain for $u < 0$ is the gain for the opposing muscle group activation. QH-Manual represents the manually tuned PI controller, and QH-PI 1 represents the PI controller tuned by GA with the same FSC of QH-Manual. QH-PI 2 represents the PI controller tuned by GA whose FSC used the muscle group of the opposing leg to decrease the velocity faster.

	QH-Manual	QH-PI 1	QH-PI 2
K_p	2.0	4.10	1.07
K_i	1.0	1.04	0.38
RQ angle interval ($^\circ$)	[250.00, 90.00]	[323.72, 367.61]	[237.001, 42.02]
RH angle interval ($^\circ$)	[20.00, 220.00]	[82.28, 143.49]	[66, 23, 226.38]
gain for $u < 0$	-	-	0.9294

5.3.1.3 Results and analysis

Simulation results of the manually tuned PI controller (QH-Manual) and the PI controller tuned by GA (QH-PI 1) are in Figure 5.10. It is possible to see, especially in the simulations without environment modifications (step input) and with noise, that the activation of QH-Manual muscles was not equally distributed between the legs. It used more the left quadriceps and right hamstrings in t_2 . In a real test, this activation distribution could accelerate the fatigue of a muscle group of one leg, and it could be difficult to control both legs with the same control signal. Also, the activation of QH-PI 1 was not equally distributed for the simulation with noise, and after 22 s the simulation stopped to work correctly. On the other hand, the bar charts from Figure 5.11.a and 5.11.c show that the activation of QH-Manual was equally distributed between quadriceps and hamstrings, while QH-PI 1 used more the hamstrings. Note that the nonzero average quadriceps activation of QH-PI 1 without environment modifications and with torque (crank load) was similar to the values of QH-Manual. However, the nonzero average hamstrings activation of QH-PI 1 without environment modifications and with torque were higher than the values for QH-Manual. That means QH-PI 1 applies more force using hamstrings. We did not compare simulations with noise because the simulation stopped to work correctly for QH-PI 1.

The outcome of different muscle recruitments can be seen in the bar charts of Figure 5.12. As the increase of activation was not high in simulations without environment modifications, QH-PI 1 had smaller maximum errors and RMSEs than QH-Manual in t_1 and t_2 . However, the increase of activation was too high in simulations with torque. Note that simulations with torque for QH-Manual had the smallest maximum errors and RMSEs, which were $3.31^\circ/s$ and 0.03 rad/s for t_1 , and $9.40^\circ/s$ and 0.06 rad/s for t_2 , respectively.

The overshoot was greater than $50.0^\circ/s$ for all simulations, but QH-PI 1 had overshoot close to $100.0^\circ/s$ (see Figure 5.12.f). In the simulation results of Figure 5.10, it is possible to see that QH-Manual used right quadriceps and left hamstrings at the beginning, while QH-PI 1 used only right quadriceps. Rise time was 0.04 s for all simulations.

The fitness function of the GA evaluated only results of simulations with step input and without environment modifications. Then, QH-PI 1 had a better performance in these conditions. The simulation stopped to work properly with the addition of noise, and simulation with torque had higher maximum error and RMSE. Also, the different angle interval for muscle activation increased the overshoot. Hence, QH-Manual was more robust when we contemplate simulations without modifications, with noise, and torque. However, the activation of QH-Manual was not equally distributed.

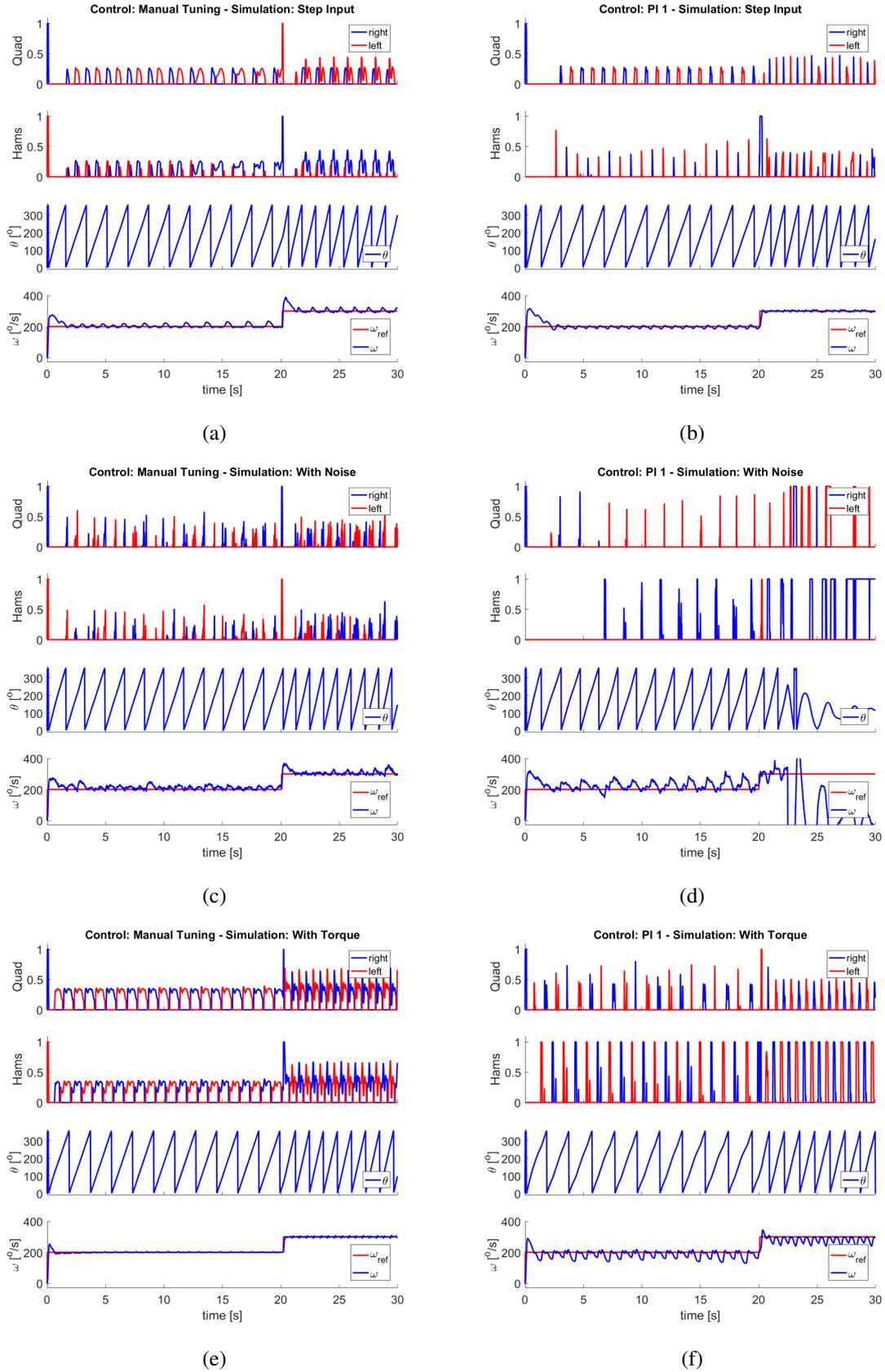


Figure 5.10: FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are a PI controller (a, c, e) manually tuned and (b, d, f) tuned by GA (PI 1). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

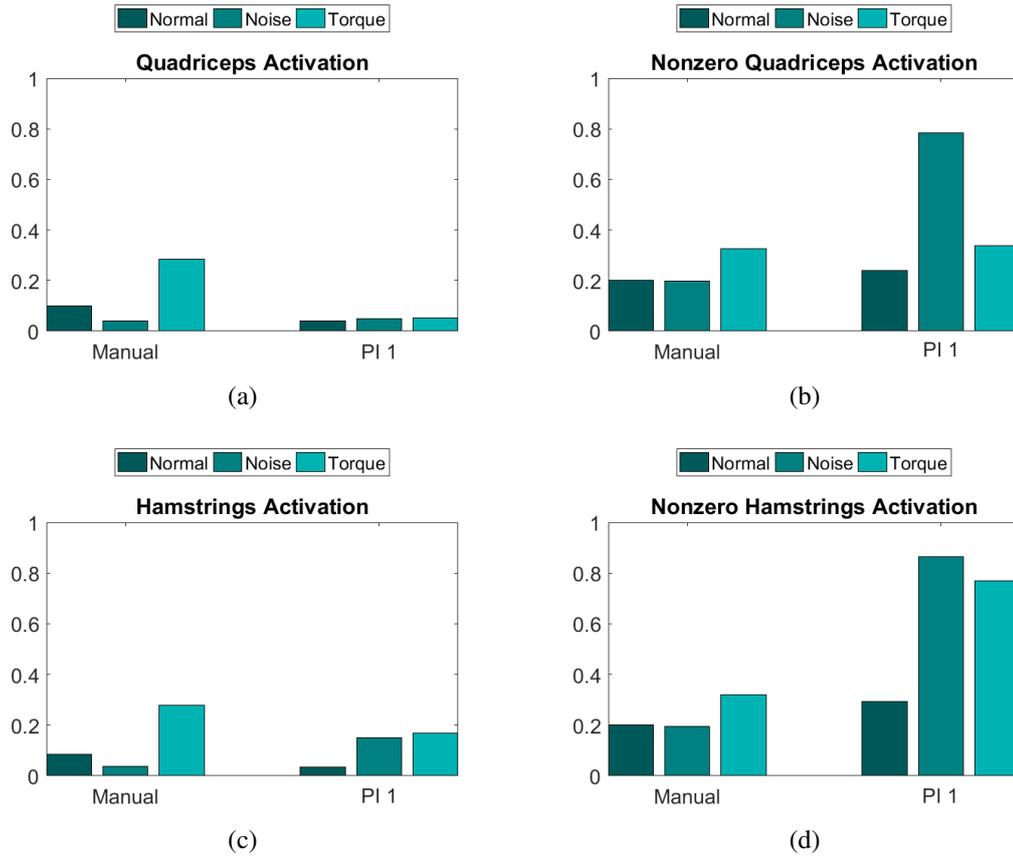


Figure 5.11: Bar charts of the controllers activating quadriiceps and hamstrings. The controllers are a PI controller manually tuned (Manual) and tuned by GA (PI 1). (a) is the average quadriiceps activation, (b) is the nonzero average quadriiceps activation, (c) is the average hamstrings activation, and (d) is the nonzero average hamstrings activation.

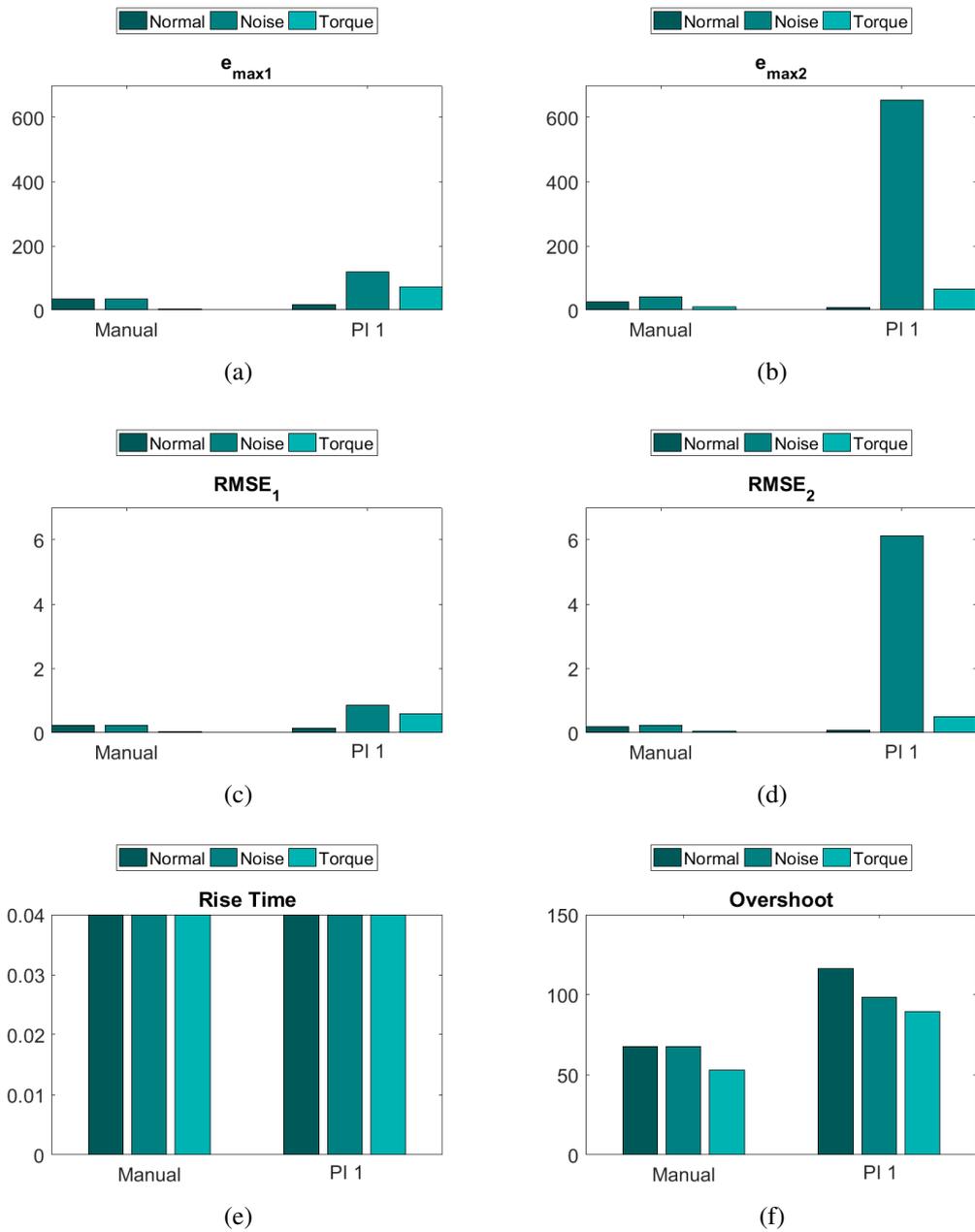


Figure 5.12: Bar charts of the controllers activating quadriceps and hamstrings. The controllers are a PI controller manually tuned (Manual) and tuned by GA (PI 1). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.3.2 Using opposing muscles to reduce velocity

5.3.2.1 Method

In past experiments with PI controller, we noticed the maximum error was smaller in simulations with torque (crank load) (see Figure 5.10.e). Since the controller is disabled when the velocity keeps higher than the desired velocity, simulations with other environment modifications had a high peak velocity, and the velocity decreased slowly. However, simulations with torque had a small peak velocity, and velocity decreased faster because of the opposing force.

We tried to add this feature to the controller. An opposing force can be obtained activating the same muscle group of the opposing leg, i.e., one leg will try to move clockwise, and the other counterclockwise. Hence, we added an opposing muscle group activation when the PI output is negative. However, the activation of the opposing muscle group cannot have the same intensity to avoid significant changes in velocity. Hence, opposing muscle group activation is multiplied by a positive gain smaller than 1. We found empirically that this opposing activation has a better performance using only the quadriceps muscle group.

Additionally, we wanted to ensure that the activation was homogeneously distributed between the legs and between the different muscle groups. Hence, we modified the fitness function of the GA. Besides the sum of the absolute value of the error between the desired velocity and the crank velocity for the whole simulation, we added:

- Sum of the absolute value of the difference between activation of muscle groups of the same leg for the whole simulation;
- Sum of the absolute value of the difference between activation of the same muscle group of both legs for the whole simulation;
- Difference between the number of samples and the average number of samples with nonzero activation.

The last term minimizes the number of samples with zero activation, i.e., the FSC always tries to choose a muscle group to be activated. The controller can react faster if it is able most of the time. The parameters we found are in Table 5.2.

5.3.2.2 Results and analysis

Simulation results of the previous PI controller tuned by GA (QH-PI 1) and the new one (QH-PI 2) are in Figure 5.13. It is possible to see at the beginning of simulations that the left quadriceps of QH-PI 2 overlaps the right quadriceps. Also, the bar charts from Figure 5.14 show that the activation of QH-PI 2 was equally distributed between the legs and between quadriceps and hamstrings. Note that the average quadriceps activation of QH-PI 2 without environment modifications and with torque was bigger than to the values for QH-PI 1. The same goes for hamstrings muscle group. On the other hand, the nonzero average quadriceps activation of QH-PI 2 without environment modifications and with torque kept almost unchanged, while the values for QH-PI 1 increased. This result confirmed that QH-PI 2 is operating most of the time. Additionally, the nonzero average hamstrings activation of QH-PI 2 with torque did not increase the value for QG-PI 1. Note in Figures 5.13.e and 5.13.f that the mean velocity of QH-PI 1 with torque was below the desired velocity during t_2 . Also, the peak value was near the desired velocity. The mean velocity for QH-PI 2 was near the desired velocity. We did not compare simulations with noise because the simulation stopped to work properly for QH-PI 1.

The outcome of the new recruitment strategy and fitness function can be seen in the bar charts of Figure 5.15. For simulations without environment modifications, QH-PI 2 had smaller maximum errors and RMSEs

than QH-PI 1 in t_1 and t_2 . The same occurred for simulations with torque.

In the simulation results of Figure 5.13, it is possible to see that QH-PI 2 used the right quadriceps and left hamstrings at the same time (i.e., the same that occurred with QH-Manual). After the velocity had increased enough, the opposing quadriceps was activated. Then, the overshoot was reduced to less than $50.0^\circ/\text{s}$. However, except for simulations with torque, the velocity increased to more than $300^\circ/\text{s}$ soon after. Rise time was 0.04 s for almost all simulations. It was 0.06 s for QH-PI 2 with noise.

In summary, QH-PI 2 solved the problems of QH-Manual and QH-PI 1. The muscle groups activation were well distributed, and the hamstrings activation did not increase in simulations with torque. Also, QH-PI 2 did not have problems in simulations with noise. Hence, QH-PI 2 was more robust when we contemplate simulations without modifications, with noise, and with torque. However, it still reaches high velocities at the beginning of most of the simulations.

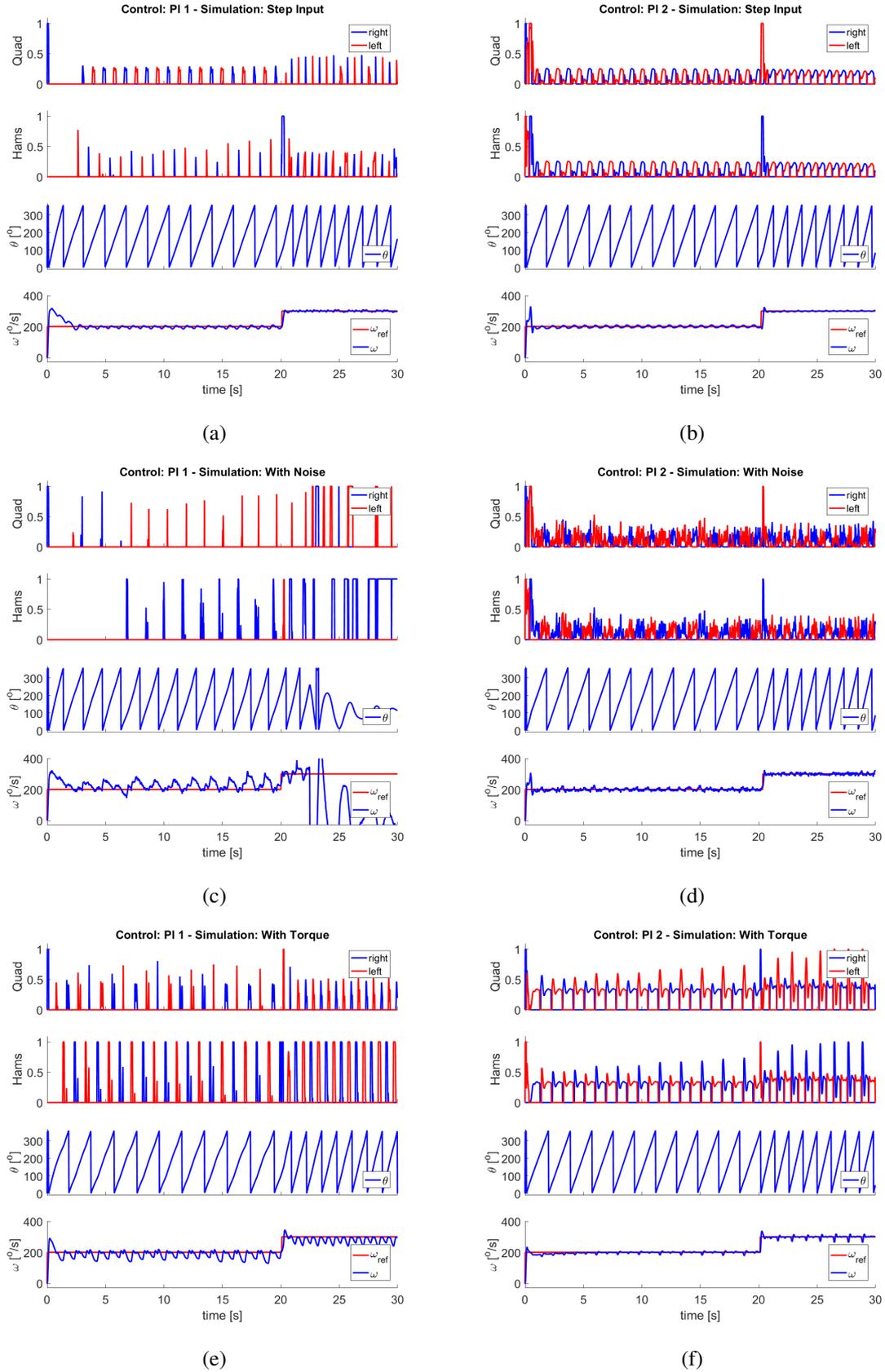


Figure 5.13: FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are (a, c, e) the previous PI controller tuned by GA (PI 1) and (b, d, f) the new one (PI 2). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

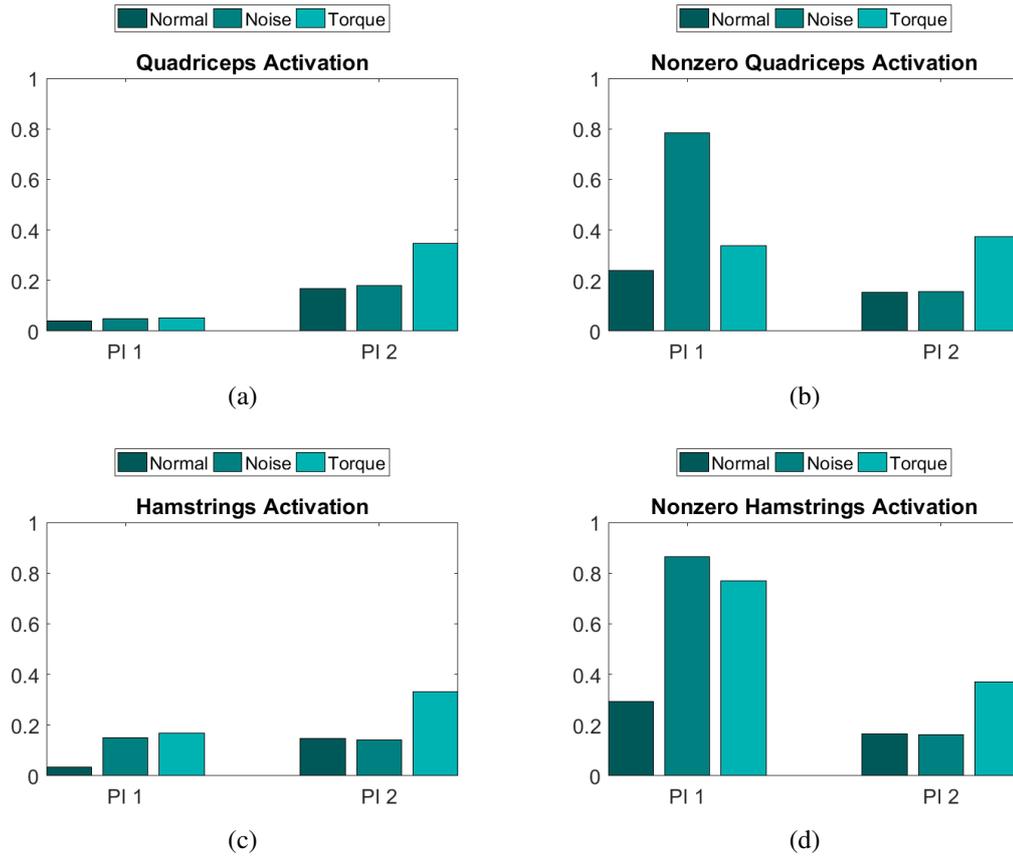


Figure 5.14: Bar charts of the controllers activating quadriceps and hamstrings. The controllers are the previous PI controller tuned by GA (PI 1) and the new one (PI 2). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, and (d) is the nonzero average hamstrings activation.

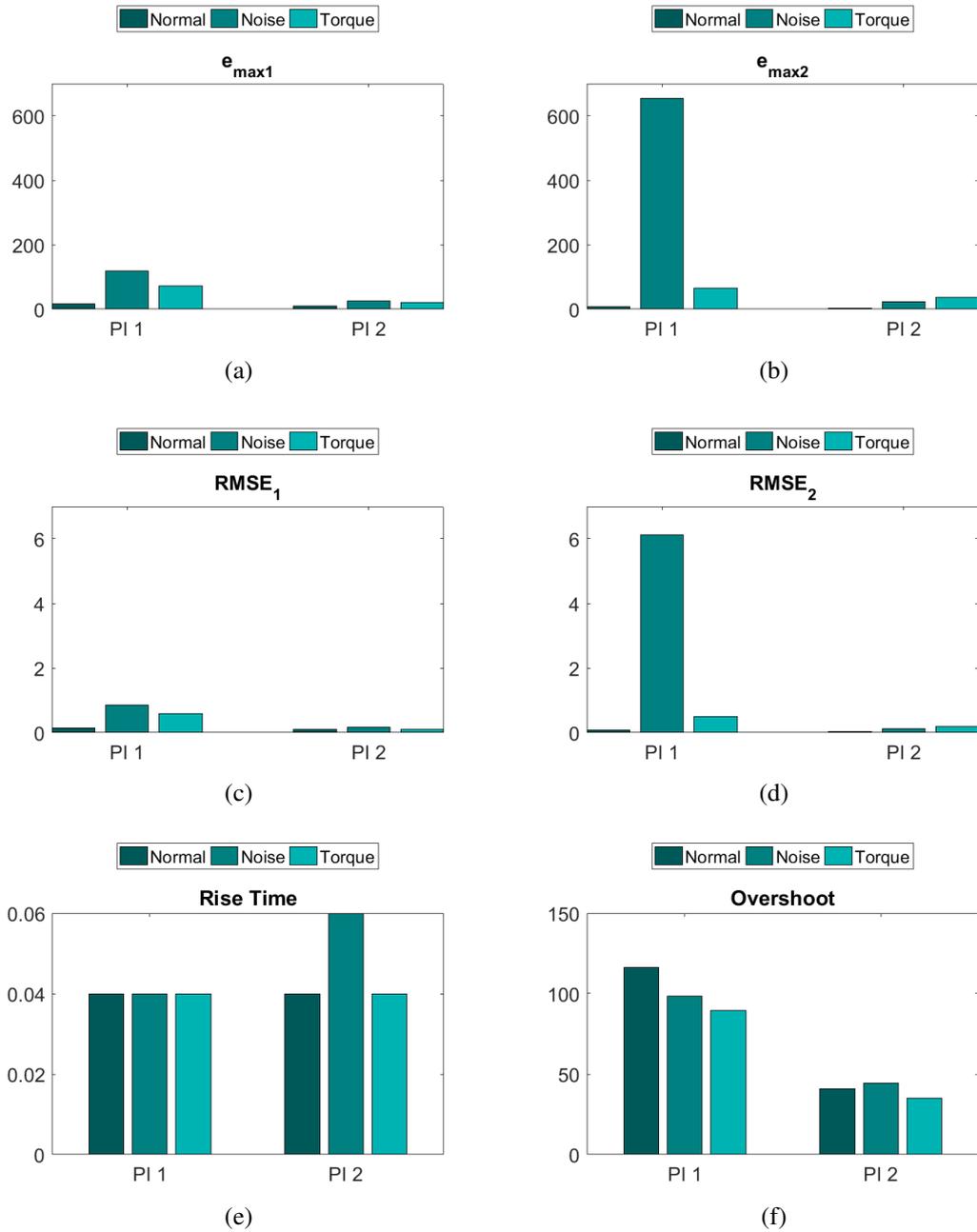


Figure 5.15: Bar charts of the controllers activating quadriceps and hamstrings. The controllers are the previous PI controller tuned by GA (PI 1) and the new one (PI 2). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.3.3 Comparison with state of the art

In state of the art, we could find FES cycling simulations of simple 2D bar linkage models (Chen et al., 1997; Gföhler and Lugner, 2004; Kim et al., 2008) to a complex 3D musculoskeletal model (Abdulla and Tokhi, 2014; Li et al., 2010; Yahaya et al., 2014). In the case of 3D models, the projects that we know use Matlab or Vn4D with controllers developed in Matlab. Besides, each model has its specific characteristics, and some used a flywheel while others used a tricycle (or just the crank). Hence, it is not easy to compare results.

Our simulations had a total time of 30 s, but the steady states t_1 and t_2 were 5-seconds length intervals. There are results for just one revolution (Abdulla and Tokhi, 2014), but a simulation time between 10 s and 15 s is most usual (Li et al., 2009; Yahaya et al., 2014; Kim et al., 2008). There are also results for more than 60 s (Chen et al., 1997; Li et al., 2010).

It is important to note that works which have used GA have presented bigger populations and more generations. Also, the probability of mutation was smaller than the value we used. Abdulla and Tokhi (2014) have initialized the population with 40 individuals, and have presented the convergence curve for 60 generations. The crossover and the mutation probabilities were 0.08 and 0.01 respectively. Additionally, the fitness function was the minimization of mean squared error in the cycling cadence. In (Yahaya et al., 2014), the GA has converged at generation 40 for a population of 20 individuals. They have used 4-seconds simulations to tune the parameters and have tested in simulations of 12 s. Kim et al. (2008) have used a population of 100 individuals, and because the computational cost of GA is high, they have used parallel computing to run the algorithm until convergence.

The simulation with the optimal parameters obtained by GA in (Abdulla and Tokhi, 2014) had a RMSE of 2.61 rpm and an error less than ± 5 rpm for the reference cadence of 35 rpm in steady state. They have developed an FLC for a humanoid-bicycle model with a flywheel and electrical clutch mechanism. Yahaya et al. (2014) have developed an FLC for a FES-assisted elliptical stepping exercise. They have obtained optimal parameters using GA and PSO. Simulation with GA parameters had RMSE of 7.87 rpm, while PSO parameters had RMSE of 7.71 rpm. The desired cadence was set to 50 rpm.

As a reference, consider the experiment conducted with an able-bodied male subject in (Bellman et al., 2014). They achieved an error less than ± 6 rpm for the desired cadence that exponentially approaches 60 rpm. They have modeled a switched system with autonomous state-dependent switching for FES cycling.

All performance measures mentioned above are from controllers with quadriceps-only activation. Chen et al. (1997) have tested a PD controller, FLC with symmetrical and asymmetrical membership functions for stimulating quadriceps and hamstrings of four participants with paraplegia. They have evaluated performance for a crank cadence of 35 rpm, 45 rpm, and 55 rpm. The mean and standard deviation of crank cadence for PD, FLC symmetric and FLC asymmetric for 35 rpm in steady state were 35.4 ± 4.8 rpm, 35.7 ± 5.4 rpm, 36.7 ± 3.4 rpm respectively. Chen et al. (2004) also have stimulated quadriceps and hamstring of a paraplegic participant, but they developed a hybrid exercise for a wheelchair with an arm crank. They obtained a cadence error bigger than ± 10 rpm for a reference of 50 rpm.

Kim et al. (2008) have done simulations with quadriceps and hamstring activation. They have developed a neuronal controller optimized by GA for a 2D bar linkage model. The desirable crank cadence was 60 rpm. The performance measures are not clear in the text, but the controller was robust enough to work with fatigue and external disturbances.

In our simulations without environment modifications, QH-PI 2 had a maximum error of $9.981^\circ/s$ (1.66 rpm) and RMSE of 0.10 rad/s (0.95 rpm) for the desired cadence of $200^\circ/s$ (33 rpm). The maximum error was $4.40^\circ/s$ (0.73 rpm) and RMSE was 0.04 rad/s (0.38 rpm) for $300^\circ/s$ (50 rpm). Even though the performance looks better than the results found in state of the art, we used OpenSim and a different tricycle design. Also, a performance without load is not realistic. QH-PI 2 had a maximum error of $20.55^\circ/s$ (3.42 rpm) and RMSE of 0.10 rad/s

(0.95 rpm) for the desired cadence of 200⁰/s (33 rpm) in simulations with torque. The maximum error was 36.62^o/s (6.10 rpm) and RMSE was 0.19 rad/s (1.81 rpm) for 300⁰/s (50 rpm). In the future, we expect to test the same control strategy in the adapted tadpole trike (HP3, Brazil), then a comparison of the performances will show if simulation results are reliable.

5.3.4 Addition of gluteus muscle group

5.3.4.1 Method

Activating just the quadriceps muscle group during FES cycling is possible. The trajectory is circular; then a leg is in the position to start extension when the extension of the other leg is over. However, this strategy demands much effort of the legs. Hence, it is most common use the hamstring muscles to flex one leg while the other is extending. Some researchers also used the gluteus muscle group to improve performance (Berkelmans, 2008; Hunt et al., 2012; Fornusek et al., 2013). In a previous work, we tested the three strategies for muscle recruitment, and simulations with quadriceps, hamstrings and gluteus muscle groups had better results (de Sousa et al., 2016).

We added gluteus muscle group to QH-PI 2 and used GA to tune the parameters. We found empirically that the opposing activation has a better performance using quadriceps and gluteus. The values of the parameters for the new PI controller (QHG-PI) are in Table 5.2.

Other types of controllers were also tested. The first was a PID controller with the derivative term (QHG-PID). However, this controller was developed before the addition of opposing muscles to reduce speed. The FSC is similar to the one used in QH-Manual and QH-PI 1, but with gluteus activation. The values of the parameters for QHG-PID are in Table 5.2.

We also tested an FLC adapted from Abdulla et al. (2014). The general method of FLC is provided by Lee (1990). The FLC (QHG-Fuzzy) uses as input the error between the desired velocity and the crank velocity, and the change in error obtained by subtracting the last sampling velocity from the new one. The inputs are normalized by scaling factors ($G_e = 1.3$ and $G_{\Delta e} = 0.022$) and fuzzified using a set of five equally distributed Gaussian membership functions. Then the algorithm infers an action according to the Fuzzy rules from Table 5.4. At last, the activation level is sent to the musculoskeletal model after defuzzification (see review in (Lee, 1990)). The scaling factors were the same from de Sousa et al. (2016). These values were already robust, and FLC is a model-free controller. The angle intervals for muscle activation were from QHG-PI because GA has a high computational cost.

At last, we decided to take advantage of the rhythmic pattern of the application and used coupled oscillators to improve the PI controller (QHG-PI+OSC). A third oscillator with a constant angular velocity equal to the desired velocity was added to ensure that the model will follow the desired velocity. Then, instead of the desired velocity, we used the output of the oscillator of the controller to calculate the error. The PI controller uses this error to generate the activation signal. The coupling constant of the controller ($a_{c1} = 1.42$) and the gain of the opposing activation (gain for $u < 0 = 0.75$) were found using GA following the framework described in Section 4.4, and the fitness function of QH-PI 2. The PI parameters and angle intervals were the same of QHG-PI, so we can compare the effectiveness of adding coupled oscillators.

5.3.4.2 Results and analysis

QHG-PI and QHG-PI+OSC results are in Figure 5.16. QHG-PID and QHG-Fuzzy results are in Figure 5.17. It is possible to see that the distribution of muscle activation between the legs for QHG-PI and QHG-

Table 5.3: The parameters for QHG-PI and QHG-PID. K_p , K_i and K_d are the proportional, integrative and derivative constants of the PI control. The angle interval is the crank angles when a muscle is activated. RQ is right quadriceps, RH is right hamstrings, and RG is the right gluteus. The muscle groups of the left leg have an angle interval equal to the angle interval of the right leg delayed 180° . The gain for $u < 0$ is the gain for the opposing muscle group activation.

	QHG-PI	QHG-PID
K_p	0.35	0.67
K_i	0.31	1.01
K_d	-	0.07
RQ angle interval ($^\circ$)	[303.00, 41.02]	[280.24, 27.17]
RH angle interval ($^\circ$)	[60.71, 233.85]	[38.38, 203.07]
RG angle interval ($^\circ$)	[54.38, 109.29]	[19.13, 110.22]
gain for $u < 0$	0.59	-

Table 5.4: Fuzzy rules for QHG-Fuzzy.

		Δe				
		NB	NS	Z	PS	PB
e	NB	NB	NB	NB	NS	Z
	NS	NB	NB	NS	Z	PS
	Z	NB	NS	Z	PS	PB
	PS	NS	Z	PB	PB	PB
	PB	Z	PB	PB	PB	PB

PI+OSC are equilibrated. QHG-Fuzzy and QHG-PID recruited more the right hamstrings and the right gluteus for the desired velocity of $300^\circ/s$, especially in simulations without environment modifications and with noise. Note that in simulations with torque, hamstrings group of both legs kept activating alternately almost all the time for all controllers, while quadriceps and gluteus groups did not. In QH-PI 2, quadriceps group of both legs kept activating alternately almost all the time. Hence, quadriceps and gluteus were sharing the function of extensor. The bar charts of Figure 5.18 show this characteristic because of the difference between the average activations and nonzero average activations.

In general, the difference between the average activations and nonzero average activations of QH-PID was greater than the other controllers, i.e., it spent more time without activating a muscle group. The same happened with QH-Manual and QH-PI 1, which also did not use the opposing muscle to reduce velocity. The biggest difference for QHG-PID was in simulations with noise (see bar charts of Figure 5.18). QHG-Fuzzy also had a significant difference between the average activations and nonzero average activations in simulations with noise.

QHG-PI and QHG-PI+OSC activated different muscle groups with similar intensities. For example, in simulations without environment modifications for QHG-PI+OSC, the nonzero average activation was 0.18 for quadriceps, hamstrings, and gluteus. QHG-PI+OSC had a slight better activation balance than QHG-PI when we contemplate simulations without modifications, with noise, and torque.

The outcome of the addition of gluteus muscle group and new control strategies can be seen in the bar charts of Figure 5.19. QHG-PI+OSC had close or better maximum errors and RMSEs without environment and with torque than QHG-PI in t_1 and t_2 . However, in the interval t_2 of simulations with noise, these measures were higher for QHG-PI+OSC than QHG-PI. It is important to note that the maximum error and RMSE for QHG-PI+OSC used the desired velocity and not the output of the controller oscillator.

QHG-PID had high maximum errors and RMSEs values with noise (see bar charts of Figure 5.19). This characteristic was also observed in QH-PI 1, but the simulation did not stop to work correctly. On the other hand, QHG-PID had low maximum errors and RMSEs in simulations with torque, which also occurred in QH-Manual. QHG-Fuzzy also had high values in simulations with torque, and it did not reach the desired velocity (see Figure 5.17.f).

The rise time was around 0.04 s and 0.06 s for the controllers except for QHG-PID. The highest value was 0.35 s for QHG-PID in simulations with torque. QHG-PI and QHG-PID did not have overshoot (see Figure 5.19.f). However, for simulations with noise, the velocity of QHG-PID increased to more than $250^{\circ}/s$ soon after. QHG-Fuzzy had a high overshoot value, while QHG-PI+OSC had a modest overshoot lower than $30^{\circ}/s$. It is possible to see in Figure 5.16.f that the red line started at 0° , i.e., the ideal controller assumed that the musculoskeletal model starts with the right leg upward. However, the right leg started positioned before the upward position (blue line). As a consequence, the coupled oscillators increased the output velocity so the model could reach the “desired angle.” QHG-PI+OSC could not have overshoot if we consider the output of the controller oscillator instead of the desired velocity.

QHG-PI+OSC is the only tested strategy that tried to follow the “desired angle.” Figure 5.16 shows that it achieved the coupling with all environment modifications. Even though this characteristic is unique and robust, it does not seem to be relevant to tests with people with SCI. Yahaya et al. (2014) have mentioned the delay observed with the strategies they tested, but they did not evaluate the consequences of this delay in the performance. Also, sometimes the actual crank angle is in advance (see Figure 5.17.d). As far as we know, there is no evaluation in state of the art of the ability to follow the “desired angle.”

QHG-PID had a performance which resembles the performance of QH-Manual and QH-PI 1, i.e., the addition of the activation of opposing muscles contributed to improving the performance of control strategies. However, it is important to note that QHG-PID and QH-Manual with torque had a better performance measure, which was not possible to reproduce with other controllers. In general, QHG-Fuzzy was robust when we contemplate simulations without modifications, with noise, and with torque. However, QHG-PI and QHG-PI+OSC had a better performance. If we take into account that the coupling with the ideal controller is not relevant, QHG-PI is more robust for its simplicity. In the future, we will add the opposing activation muscle to PID controller and use GA to tune FLC parameters for further investigation.

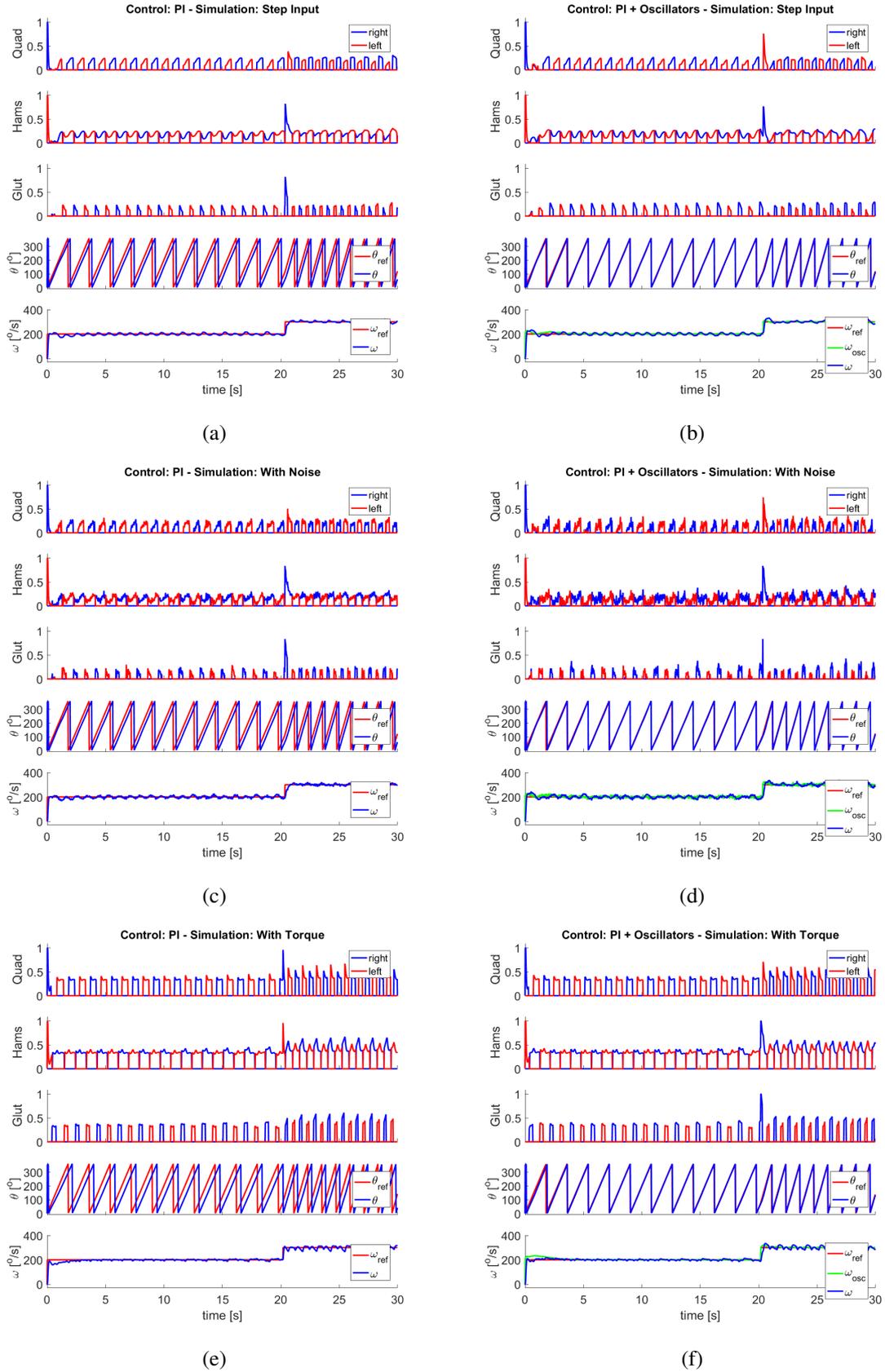


Figure 5.16: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

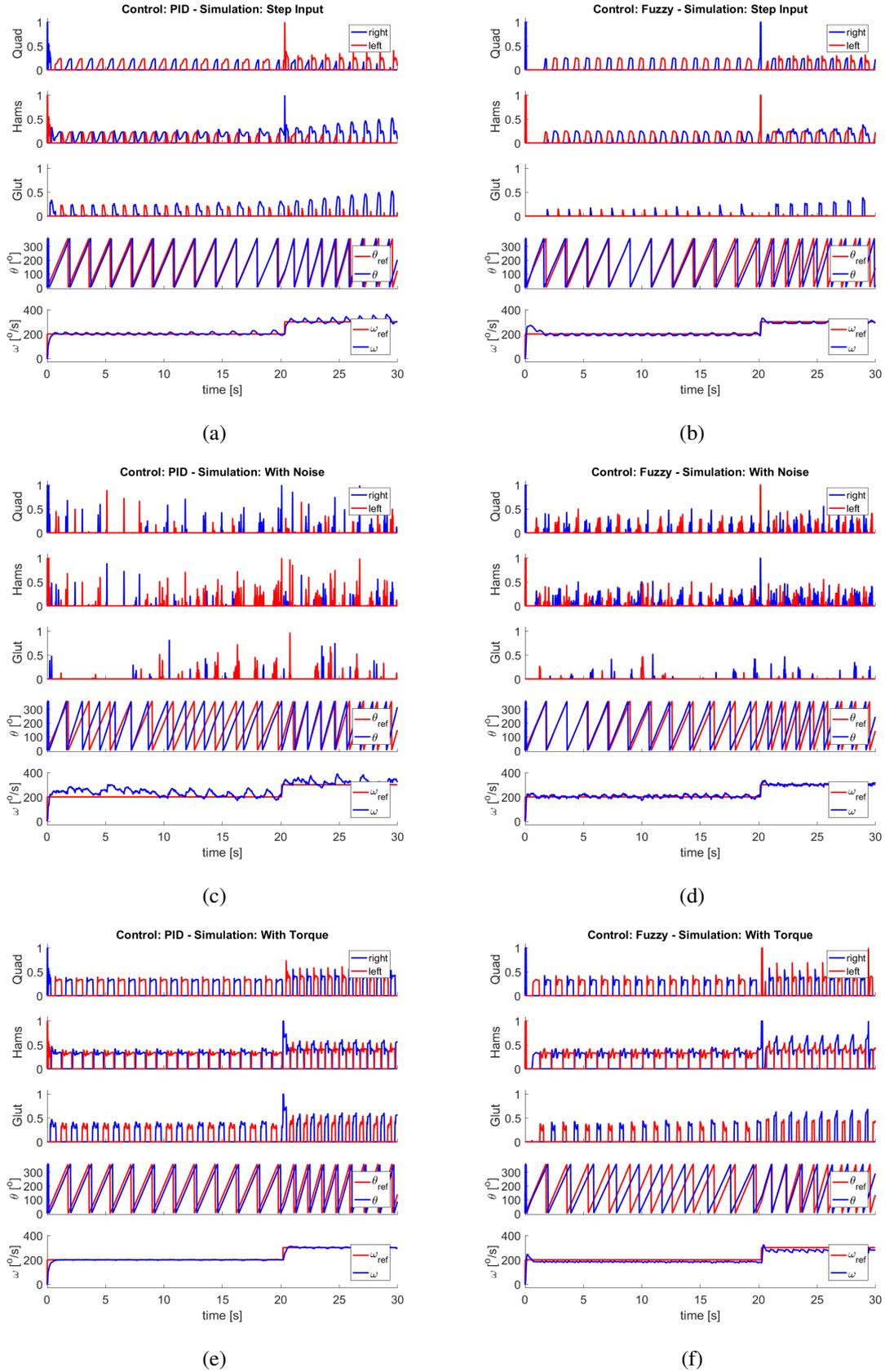


Figure 5.17: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PID controller and (b, d, f) the FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

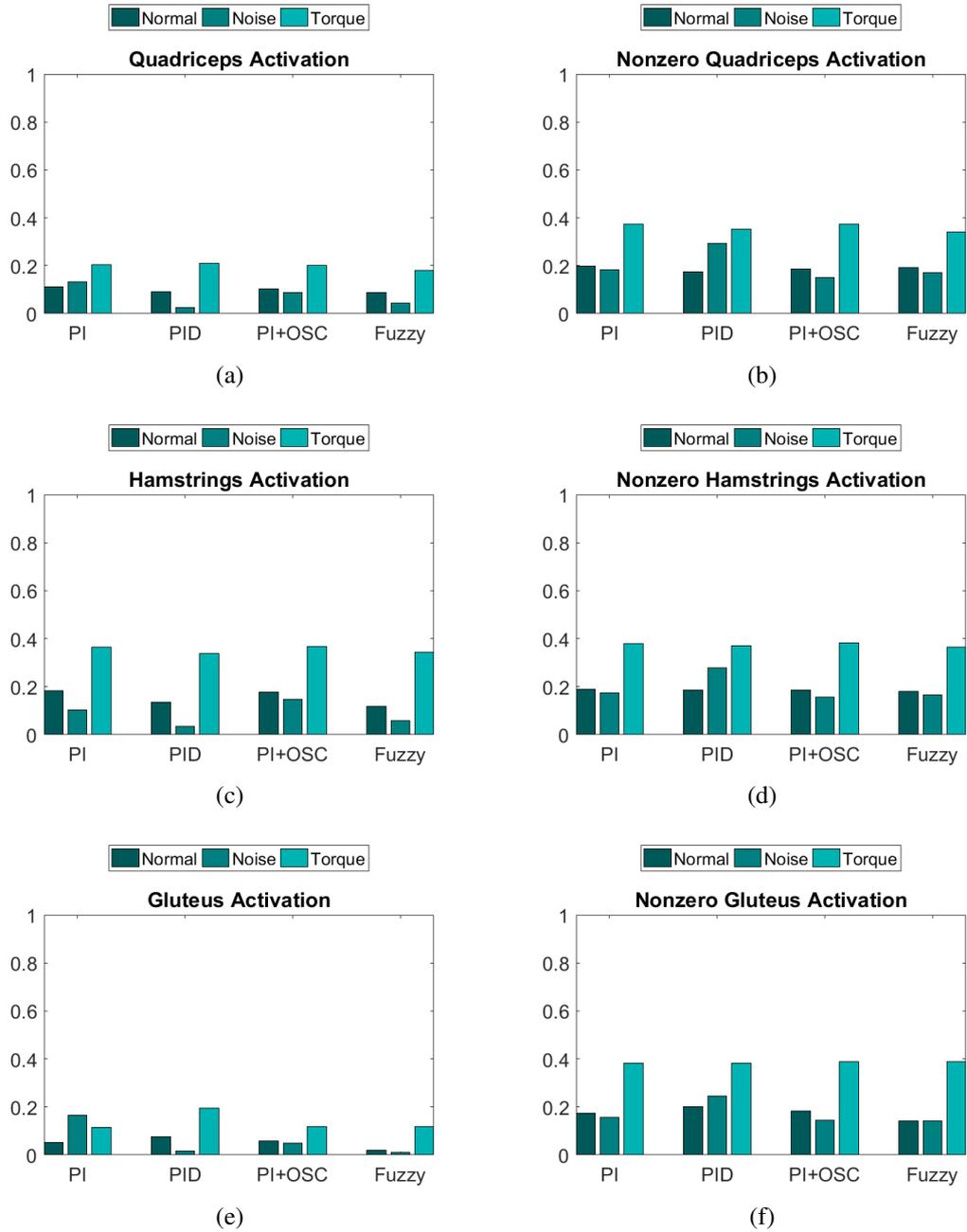


Figure 5.18: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PID, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.

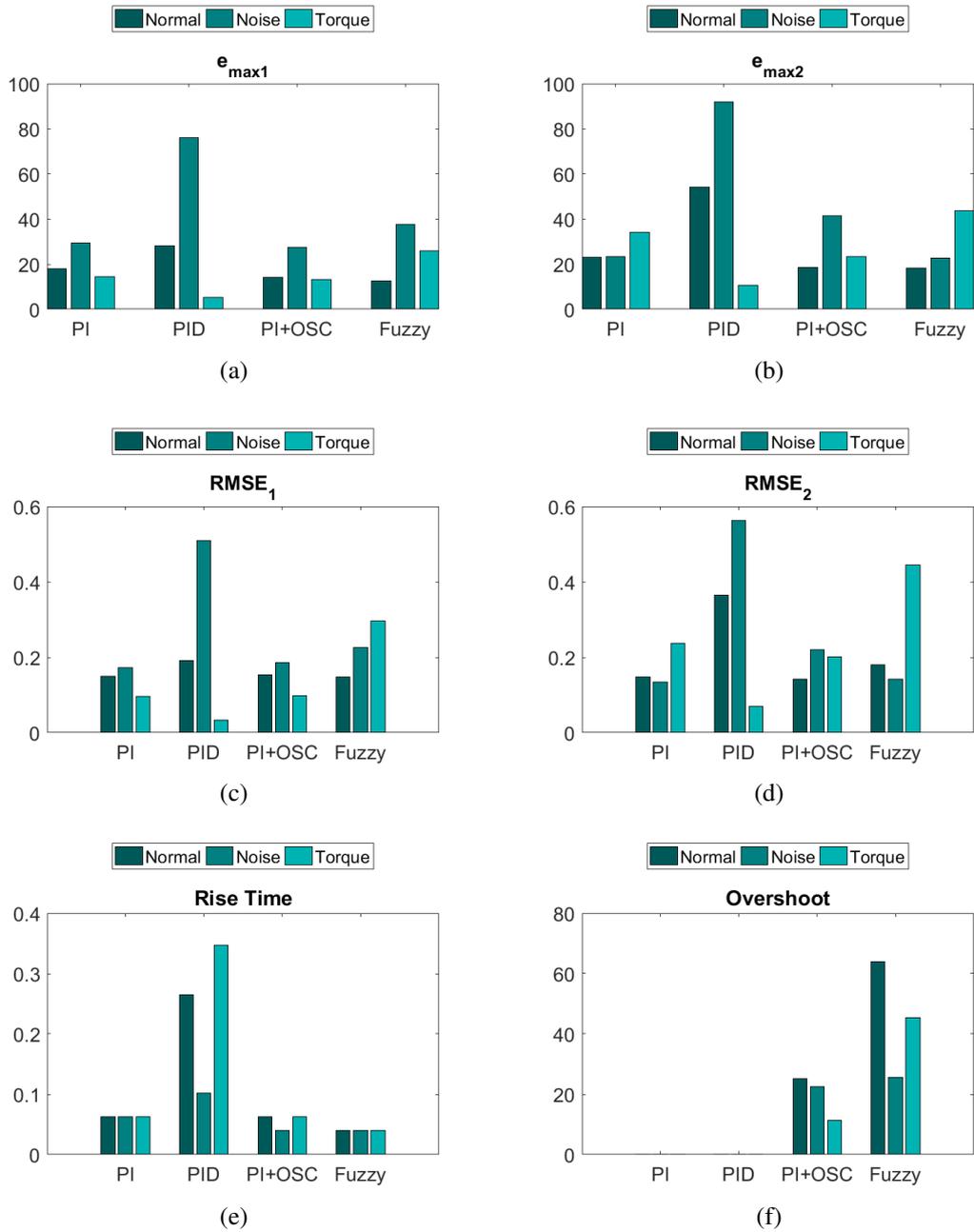


Figure 5.19: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PID, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.3.5 Study about the influence of load

5.3.5.1 Method

First, we decided to implement a simple torque against the movement of the crank to increase the effort. The torque increased proportionally to velocity, so the model did not move backward at the beginning of simulations.

The bushing force from OpenSim can be programmed to do the same thing. Controlling the load tuning the damping coefficient is possible. Additionally, OpenSim can record the force applied in the crank, so it is feasible to record the crank load.

Kim et al. (2008) applied a crank load of 10.7 N·m in their simulations with the desired velocity of 60 rpm (360°/s). They have developed a neuronal controller optimized by GA for a 2D bar linkage model. An example of torque generated by a bushing force is shown in Figure 5.20. The damping coefficient was 3 N·m/(rad/s). The torque intensity varied around 12 N·m during t_1 and it varied around 18 N·m during t_2 . We also did simulations using damping coefficient values of 1 N·m/(rad/s), 2 N·m/(rad/s), 3 N·m/(rad/s), 4 N·m/(rad/s) and 5 N·m/(rad/s).

5.3.5.2 Results and analysis

Simulation results for QHG-PI and QHG-PI+OSC are in Figures 5.21 and 5.22. Simulation results for QHG-Fuzzy are in Figures 5.23 and 5.24. It is important to notice that the parameters of controllers and starting conditions were the same used in previous simulations because we wanted to evaluate which controllers are robust enough to work with different environment modifications. We did not use QHG-PID because it did not have the activation of opposing muscles.

The bar charts from Figures 5.25 show that muscle activation increased when the damping coefficient increased, i.e., the muscles had to apply more force to overcome a bigger load. Furthermore, the difference between the average activations and nonzero average activations for quadriceps and gluteus increased when the damping coefficient increased. For example, the average quadriceps activation of QHG-PI is 0.31 for 3 N·m/(rad/s), and the nonzero average quadriceps activation is 0.62 for the same damping coefficient. The difference between the average hamstrings activations and nonzero average hamstrings activations was small. At first, it seems that the control strategies decreased the time interval when the extensors are activated but increased the intensity when we used more load in the crank. However, it is possible to see in simulation results (Figures 5.21, 5.22, 5.23 and 5.24) the mean velocity reached by the model decreased when we used more load, i.e., the crank angle spent more time outside the angle interval for quadriceps and gluteus muscle groups activation.

The outcome of the addition of different loads can be seen in the bar charts of Figure 5.26. Just as happened with muscle activation, maximum errors and RMSEs increased when the damping coefficient increased. The increasing rate of QHG-Fuzzy is more stable than other controllers, but the mean velocity reached by the model is lower for high damping coefficients values (see Figures 5.23 and 5.24). Also, the maximum errors and RMSEs of QHG-Fuzzy were bigger than the values of QHG-PI and QHG-PI+OSC. Figures 5.21 and 5.22 show that the output of the controller oscillator of QHG-PI+OSC was higher than the desired velocity so the model could reach the “desired angle.” As a consequence, the maximum errors in t_1 for QHG-PI+OSC were smaller than QHG-PI, except for damping coefficient value of 5 N·m/(rad/s). However, the model in QHG-PI+OSC could not follow the “desired angle” for 5 N·m/(rad/s). The same happened in t_2 for 4 N·m/(rad/s) and 5 N·m/(rad/s).

The rise time was around 0.04 s and 0.06 s for the controllers except for QHG-PI for damping coefficient bigger than 2 N·m/(rad/s). QHG-PI did not have overshoot (see Figure 5.26.f). QHG-Fuzzy and QHG-PI+OSC

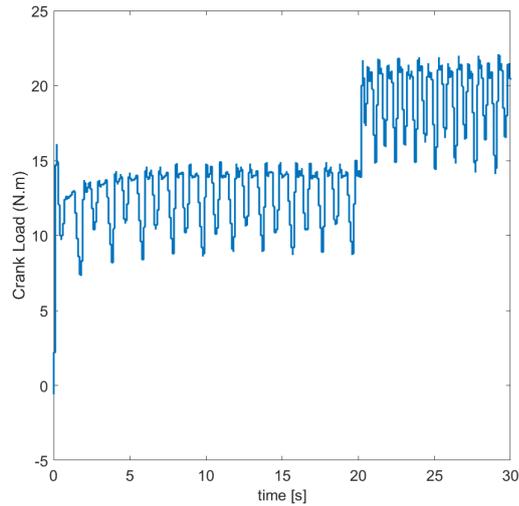


Figure 5.20: Example of torque generated by a bushing force. The torque increased proportionally to velocity.

had high overshoot values in simulations with load, but the overshoot decreased with the increase of load. It is possible to see in Figure 5.21.f that the red line started at 0° , i.e., the ideal controller assumed that the musculoskeletal model starts with the right leg upward. However, the right leg started positioned before the upward position (blue line). As a consequence, the coupled oscillators from QHG-PI+OSC increased the output velocity so the model could reach the “desired angle.”

The use of a bushing force as the crank load caused an undesired variation of torque. Hence, the results were not realistic. However, it is important to note that QHG-PI+OSC could couple with the ideal controller or follow it with a constant difference phase, except when the torque was too big. As a result, the maximum errors and RMSEs were smaller than the values obtained with QHG-PI. Additionally, Kim et al. (2008) achieved an error of approximately $\pm 120^\circ/s$ for the desired velocity of $360^\circ/s$ and a crank load of $10.7 \text{ N}\cdot\text{m}$. QHG-PI+OSC achieved a maximum error of $108.57^\circ/s$ for the desired velocity of $200^\circ/s$ and a load varying around $12 \text{ N}\cdot\text{m}$ (see Figure 5.22.b). The maximum error was $122.04^\circ/s$ for QHG-PI and $121.61^\circ/s$ for QHG-Fuzzy with the same conditions. For further investigation, we are planning to change the starting position of the model to avoid moving backward and add a constant torque opposing the movement in the crank.

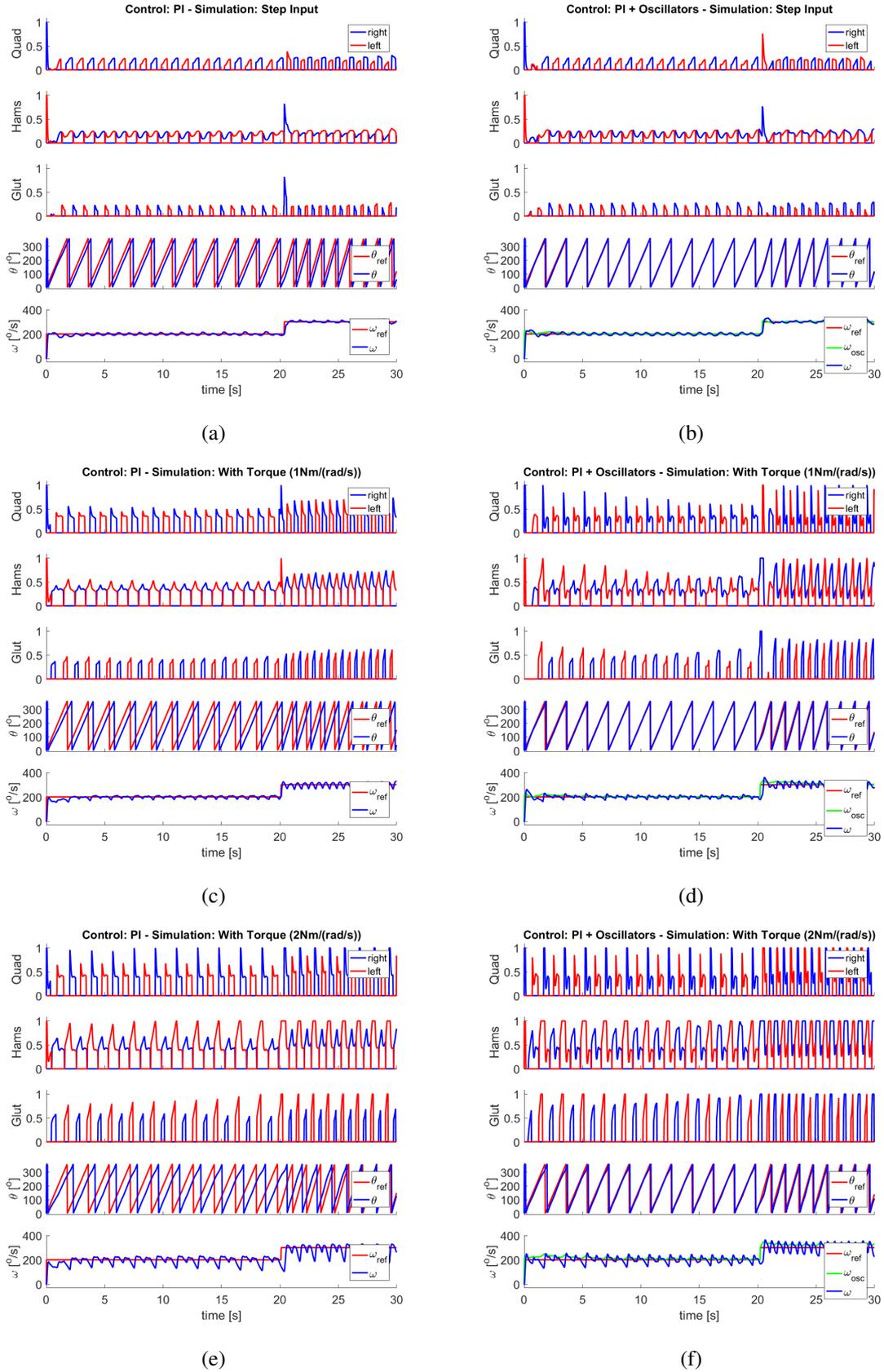


Figure 5.21: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, and with a bushing force using damping coefficient values of (c, d) $1 \text{ N}\cdot\text{m}/(\text{rad}/\text{s})$ and (e, f) $2 \text{ N}\cdot\text{m}/(\text{rad}/\text{s})$.

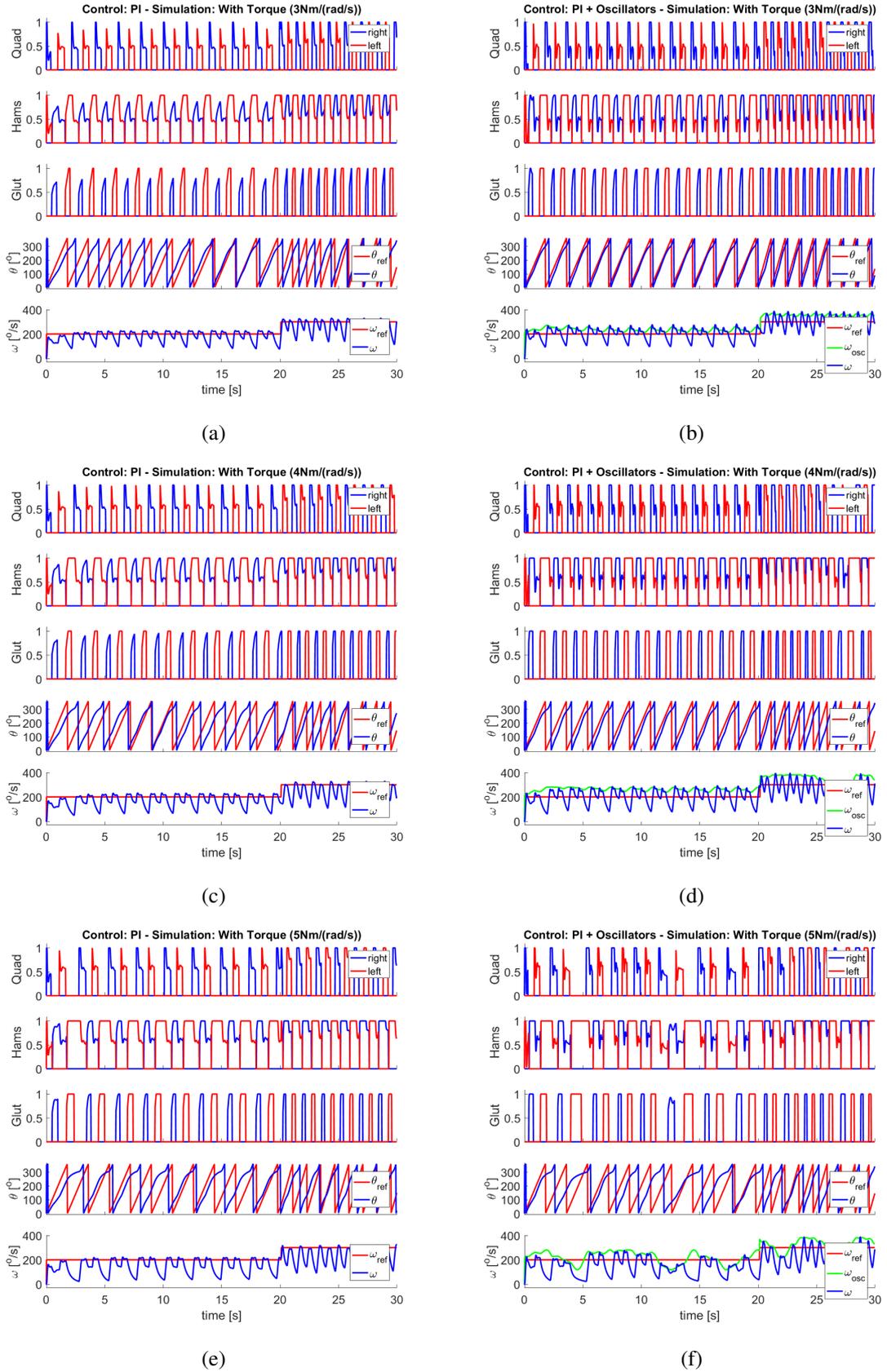
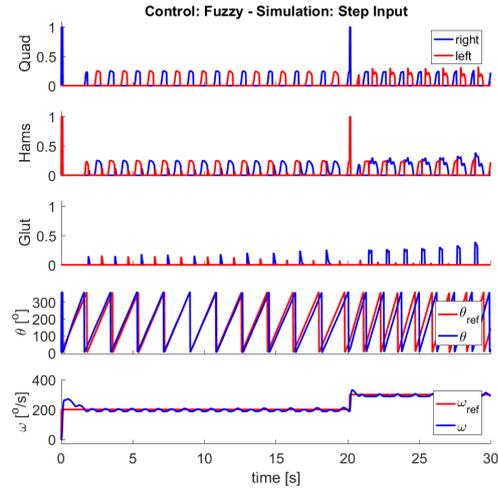
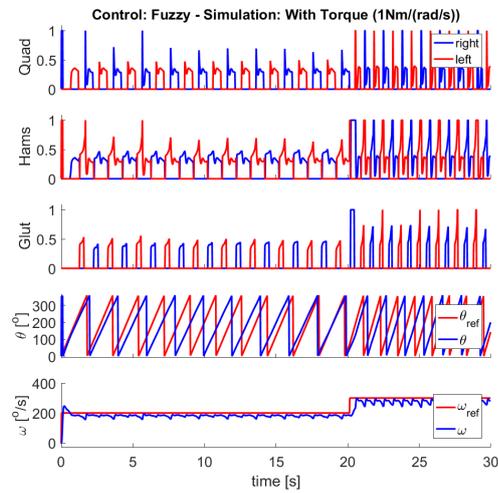


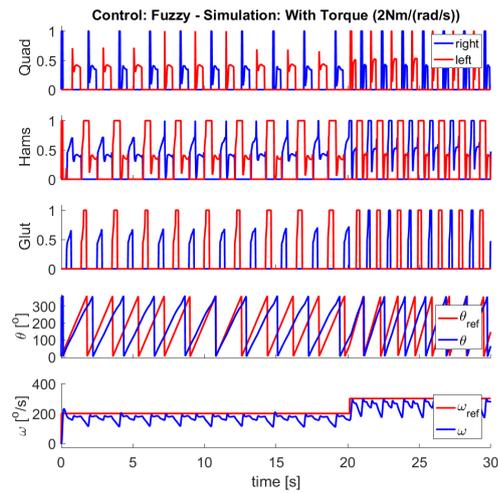
Figure 5.22: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a bushing force using damping coefficient values of (a, b) 3 N·m/(rad/s), (c, d) 4 N·m/(rad/s) and (e, f) 5 N·m/(rad/s).



(a)

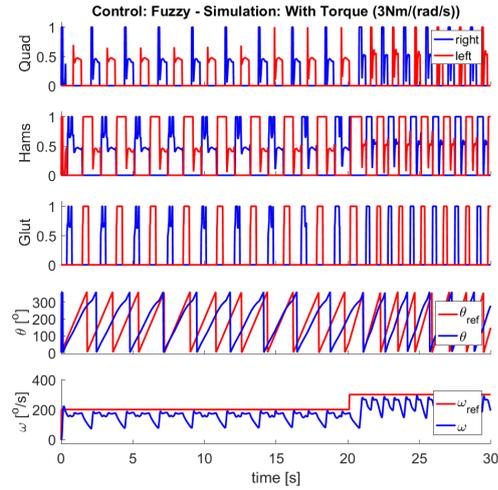


(b)

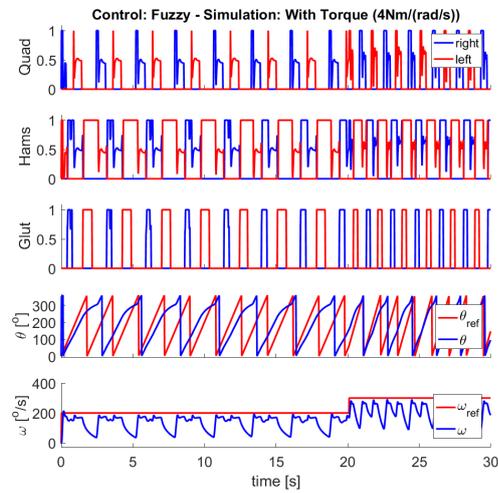


(c)

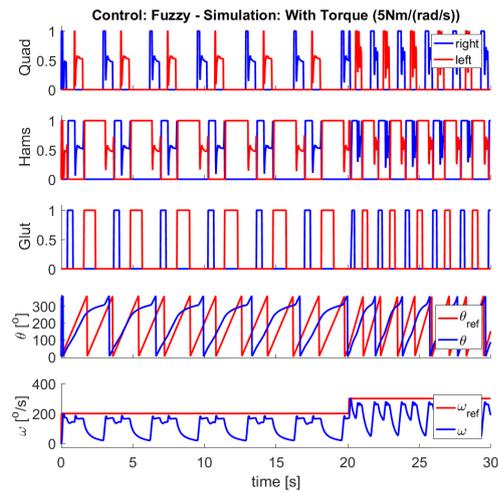
Figure 5.23: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a) without environment modifications, and with a bushing force using damping coefficient values of (b) 1 N·m/(rad/s) and (c) 2 N·m/(rad/s).



(a)



(b)



(c)

Figure 5.24: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controller is FLC (Fuzzy). We performed simulations with a bushing force using damping coefficient values of (a) 3 N·m/(rad/s), (b) 4 N·m/(rad/s) and (c) 5 N·m/(rad/s).

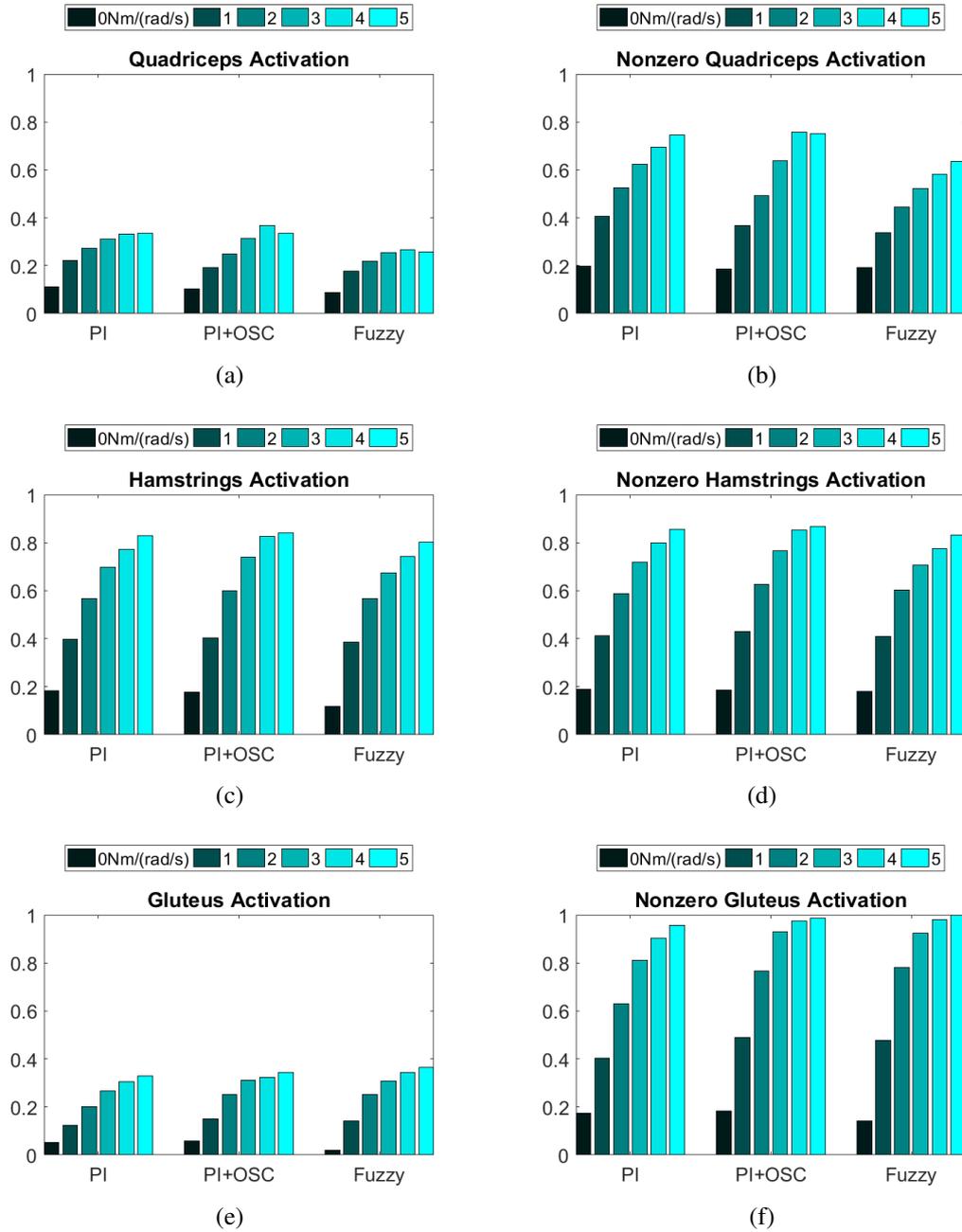


Figure 5.25: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.

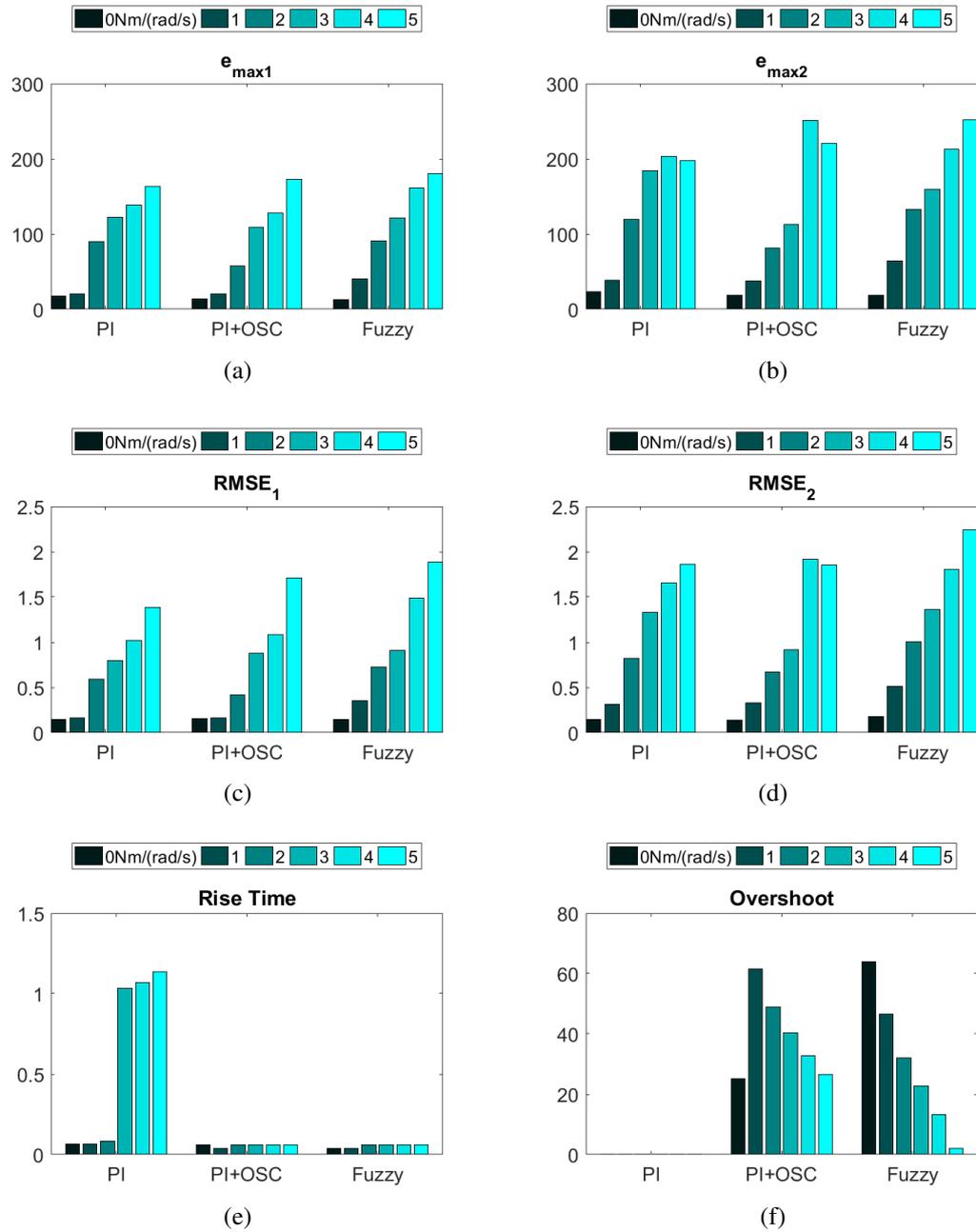


Figure 5.26: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.3.6 Different scales

5.3.6.1 Method

Besides the simplicity of the musculoskeletal model compared to a person, people may have different heights and weights. The musculoskeletal model based on Delp et al. (1990) is about 1.80 m tall and has a mass of 75.16 kg. We also wanted to test if the control strategies were robust enough with the same model but different heights. Hence, we used the Scaling Tool of OpenSim⁵ to create new scaled models. The Scale Tool compares marker data from a person with markers on the model and scales the model according to the person. However, for this work, we just scaled manually.

Two models were created. One is 1.70 m tall and other is 1.90 m tall. The pedals and crankset were not scaled since the real equipment is the same for all participants. However, we changed the initial position of the models to have smoother simulations. Tables 5.5 and 5.6 show the initial angles and which joints are locked for the models which are 1.70 m and 1.90 m tall, respectively.

5.3.6.2 Results and analysis

Simulation results for QHG-PI and QHG-PI+OSC are in Figures 5.27 and 5.29. Simulation results for QHG-Fuzzy are in Figures 5.28 and 5.30. We used the height to denote which is the model. It is important to notice that the parameters of controllers and starting conditions were the same used in previous simulations because we wanted to evaluate which controllers are robust enough to work with different scales. We did not use QHG-PID because it did not have the activation of opposing muscles.

QHG-PI(1.90) with torque stopped working correctly after 2.00 s and QHG-PI+OSC(1.90) with torque also stopped working correctly after 6.00 s. Additionally, QHG-Fuzzy(1.70) and QHG-Fuzzy(1.90) with torque did not reach the desired velocity. QHG-PI and QHG-PI+OSC had a visible instability in simulations without environment modifications, especially in t_2 .

The bar charts of Figures 5.31, 5.32 and 5.33 show that the nonzero average activation increases a lot in simulations with torque. The nonzero average gluteus activation for the model which is 1.90 m tall almost reached the maximum value for QHG-Fuzzy. The nonzero average activation of QHG-Fuzzy(1.90) without environment modifications did not change much when compared to the nonzero average activation of QHG-

⁵<https://simtk-confluence.stanford.edu:8443/display/OpenSim/Scaling>

Table 5.5: Initial angles and which joints are locked in the FES cycling model which is 1.70 m tall.

Joint	Initial Angle (°)	Locked
pelvis_tilt	44.712	yes
pelvis_tx	-0.663	yes
pelvis_ty	0.052	yes
hip_flexion_r	104.005	no
knee_angle_r	-81.732	no
ankle_angle_r	-0.000	yes
hip_flexion_l	78.655	no
knee_angle_l	-82.525	no
ankle_angle_l	0.000	yes
lumbar_extension	0.000	yes
crankSetToGround	2.720	no

Table 5.6: Initial angles and which joints are locked in the FES cycling model which is 1.90 m tall.

Joint	Initial Angle ($^{\circ}$)	Locked
pelvis_tilt	44.475	yes
pelvis_tx	-0.768	yes
pelvis_ty	0.048	yes
hip_flexion_r	97.705	no
knee_angle_r	-78.804	no
ankle_angle_r	-0.002	yes
hip_flexion_l	73.640	no
knee_angle_l	-75.551	no
ankle_angle_l	0.002	yes
lumbar_extension	0.000	yes
crankSetToGround	6.755	no

Fuzzy(1.80). The same goes for QHG-Fuzzy(1.90) in simulations with noise and QHG-PI(1.90) in both cases. The nonzero average activation values of QHG-PI+OSC(1.90) without environment modifications were higher than the values of QHG-PI(1.80). On the other hand, the nonzero average activation for the model which is 1.70 m tall in general increased.

The outcome of the different scales can be seen in the bar charts of Figure 5.34, 5.35 and 5.36. In general, maximum errors and RMSEs values for QHG-PI and QHG-PI+OSC without environment modifications for the models which are 1.70 m and 1.90 m tall were higher when compared with the model which is 1.80 m tall, especially in t_2 . The maximum errors and RMSEs for QHG-Fuzzy(1.70) in simulations with noise in t_1 and t_2 were similar to the values for QHG-Fuzzy(1.80). The same was true for the model which is 1.90 m tall. Furthermore, the performance of the models which are 1.70 m and 1.90 m tall did not change much for QHG-Fuzzy in simulations with noise.

The rise time was around 0.04 s and 0.06 s for the controllers except for QHG-PI(1.70), which was around 0.10 s. The model which is 1.90 m tall was the only model which had QHG-PI with overshoot. The overshoot for QHG-PI+OSC increased for both scaled models when compared to the model which is 1.80 m tall.

In some cases, a control strategy seems to be better, while in other cases, the performance is the same regardless of scale. Instabilities in the angular velocity in some simulations may have affected simulation results, and some simulations stopped working properly. Hence, the different performances might be a result of the scaling method which did not take into account numerical problems. However, FLC had a better performance when we contemplate simulations without modifications and with noise. That means FLC may achieve better results when we use the same parameters for participants with different heights, which is an expected characteristic of a model-free controller. Also, it is important to note that QHG-PI+OSC could couple with the ideal controller or follow it with a constant difference phase.

As far as we know, no previous study simulates FES cycling with different models. However, in real tests, it is possible to find tests with various participants (Fornusek et al., 2007; Hunt et al., 2002). For further investigation, we are planning to adjust simulation parameters such as sampling time and change the ODE solver.

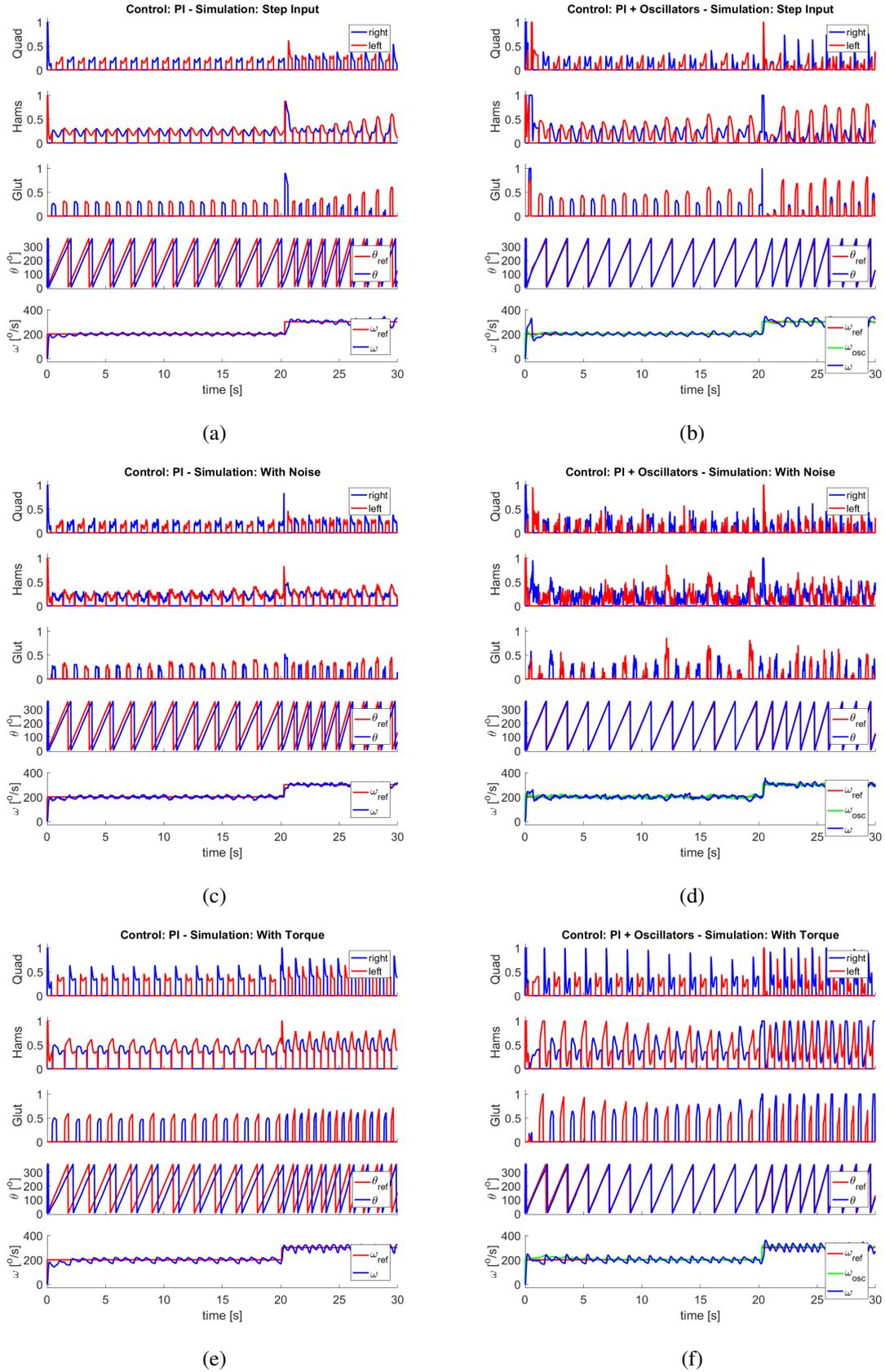
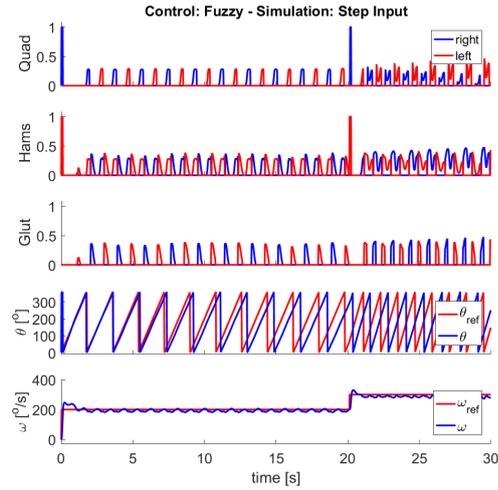
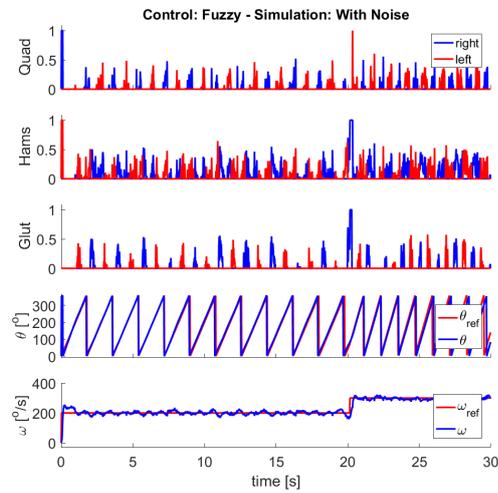


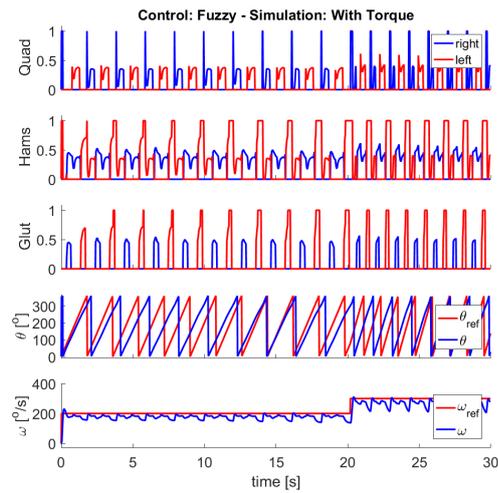
Figure 5.27: FES cycling simulation results for the model which is 1.70 m tall. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.



(a)



(b)



(c)

Figure 5.28: FES cycling simulation results for the model which is 1.70 m tall. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

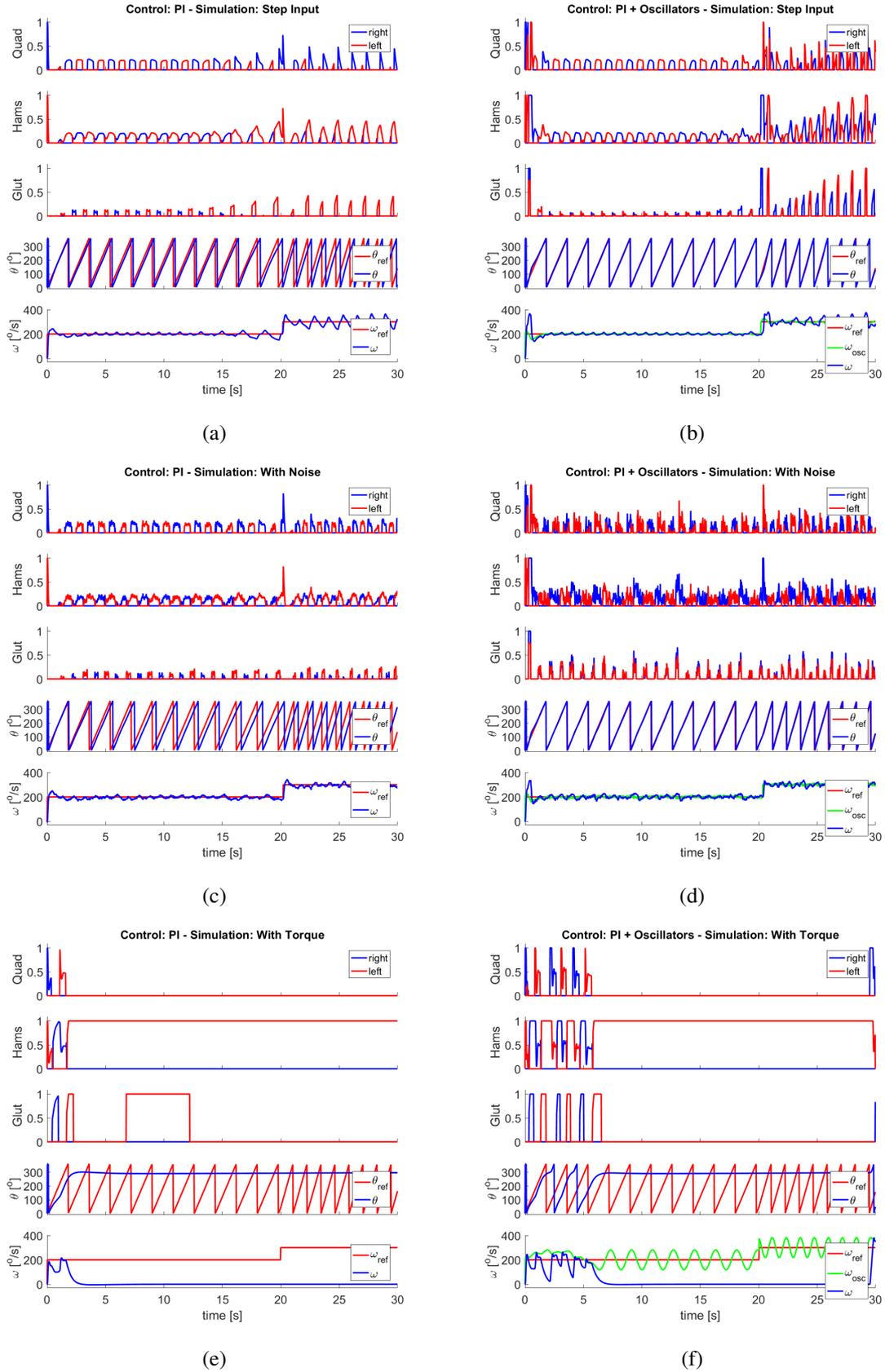
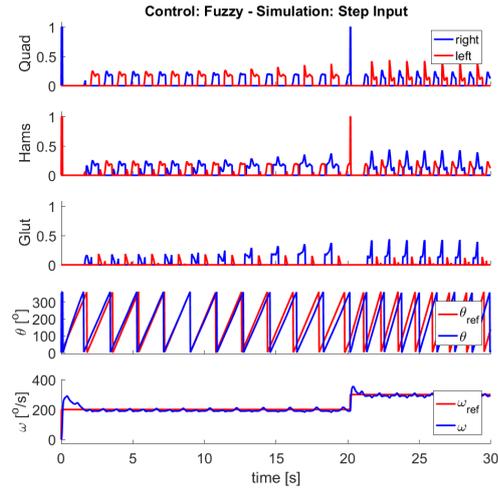
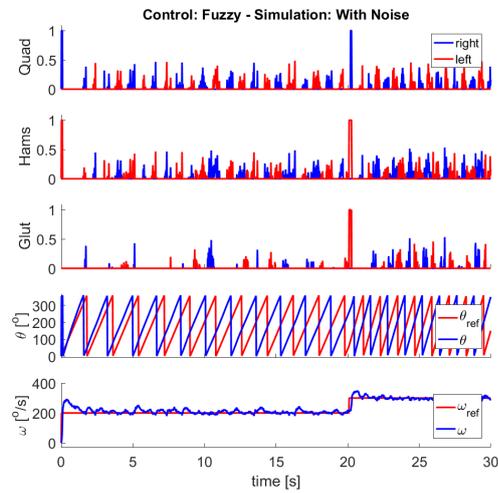


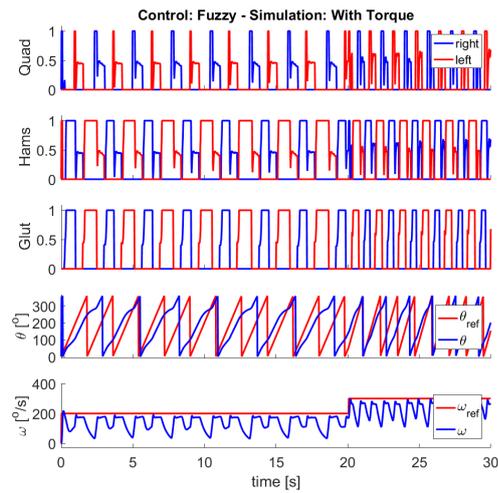
Figure 5.29: FES cycling simulation results for the model which is 1.90 m tall. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.



(a)



(b)



(c)

Figure 5.30: FES cycling simulation results for the model which is 1.90 m tall. The controller is FLC (Fuzzy). We performed simulations with a step input signal (a, b) without environment modifications, (c, d) with noise, and (e, f) torque.

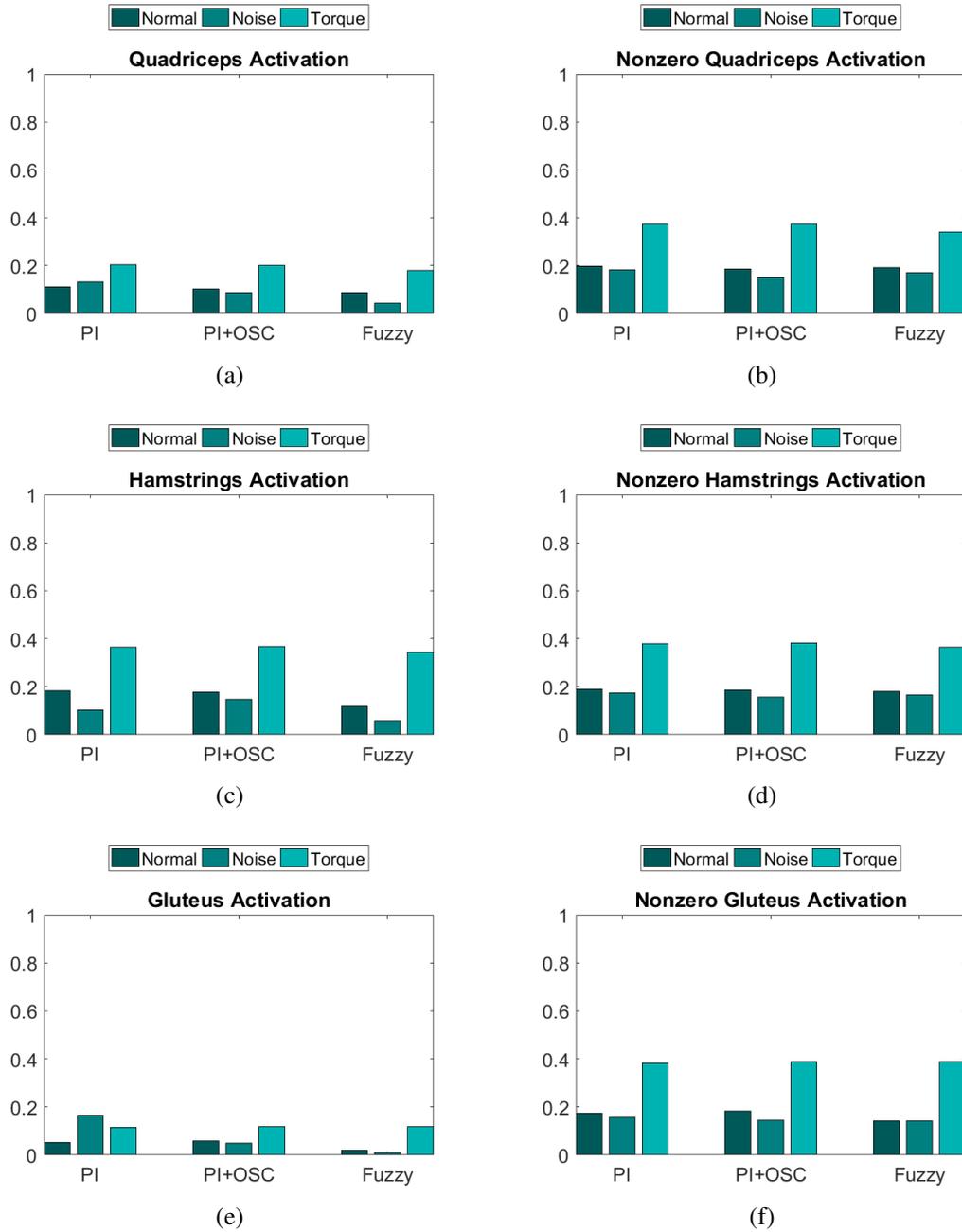


Figure 5.31: Bar charts for the model which is 1.80 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.

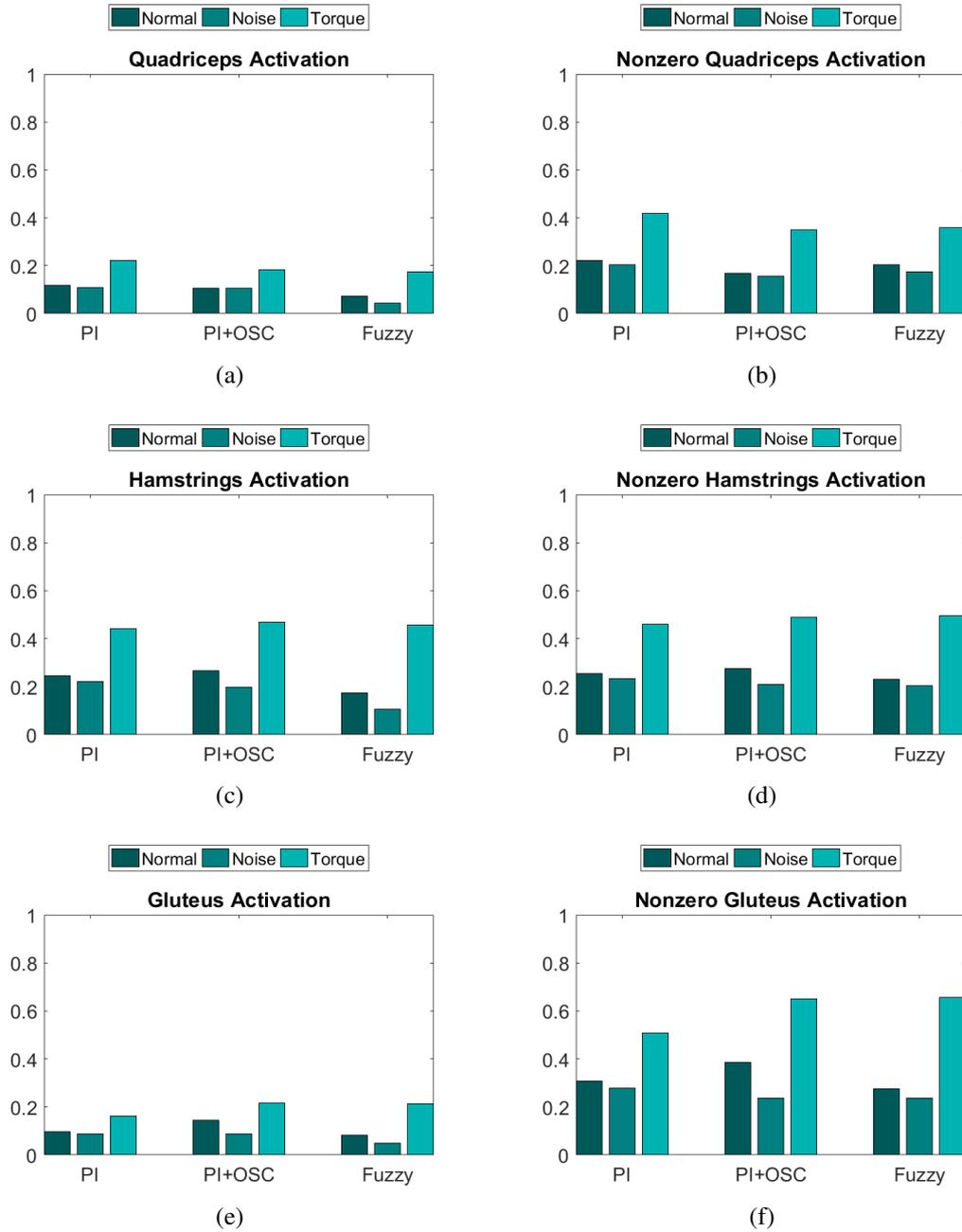


Figure 5.32: Bar charts for the model which is 1.70 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.

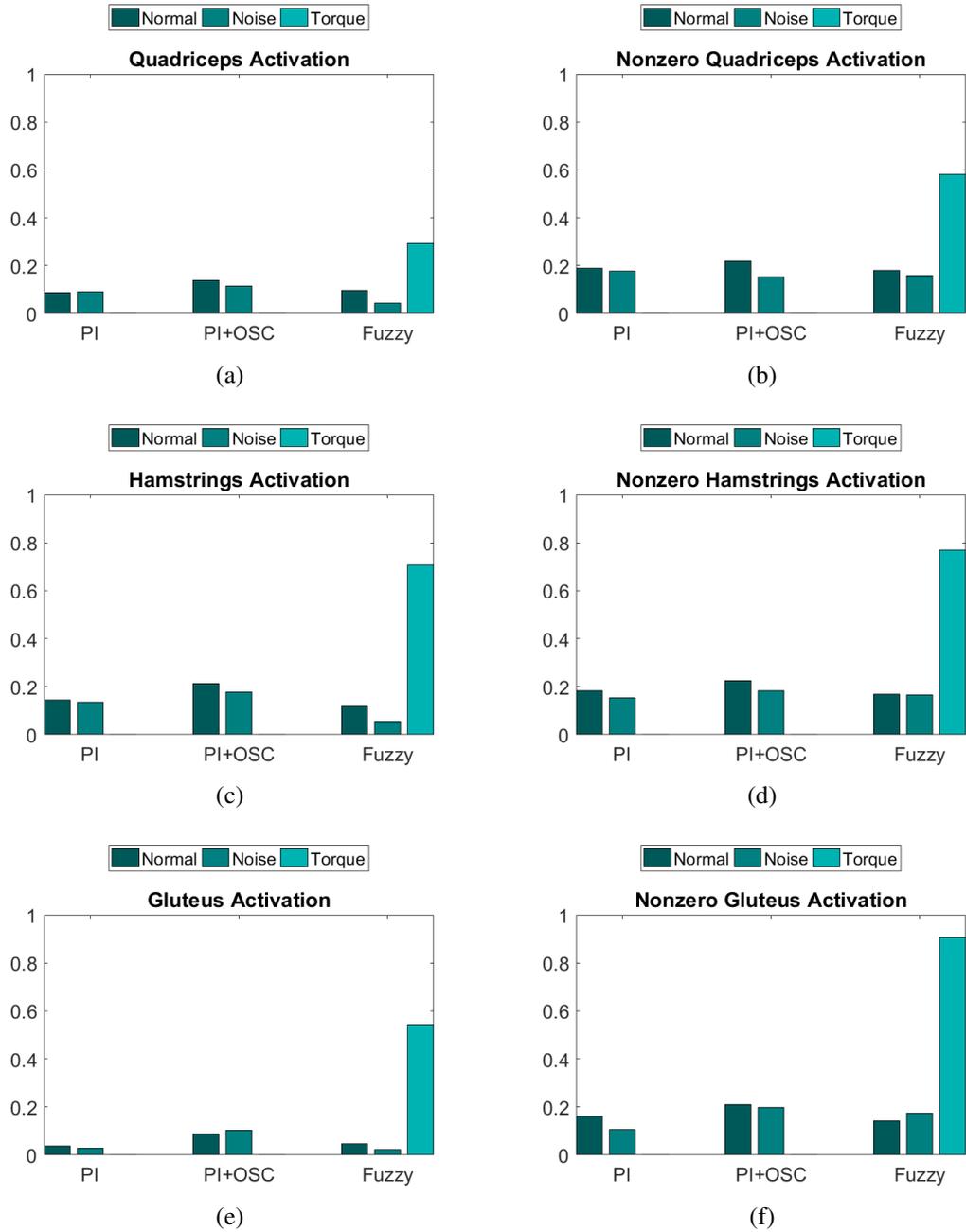


Figure 5.33: Bar charts for the model which is 1.90 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the nonzero average quadriceps activation, (c) is the average hamstrings activation, (d) is the nonzero average hamstrings activation, (e) is the average gluteus activation, and (f) is the nonzero average gluteus activation.

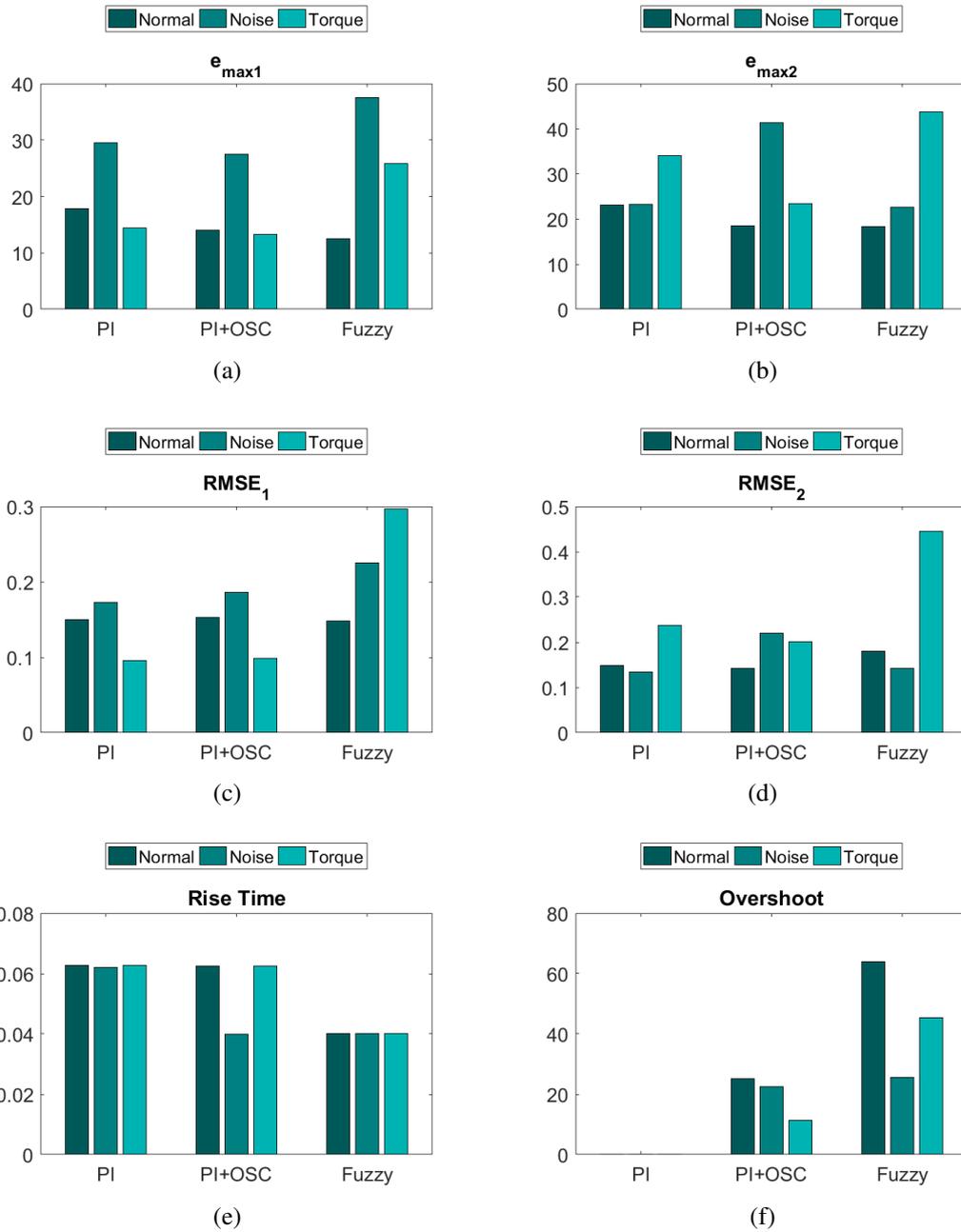


Figure 5.34: Bar charts for the model which is 1.80 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

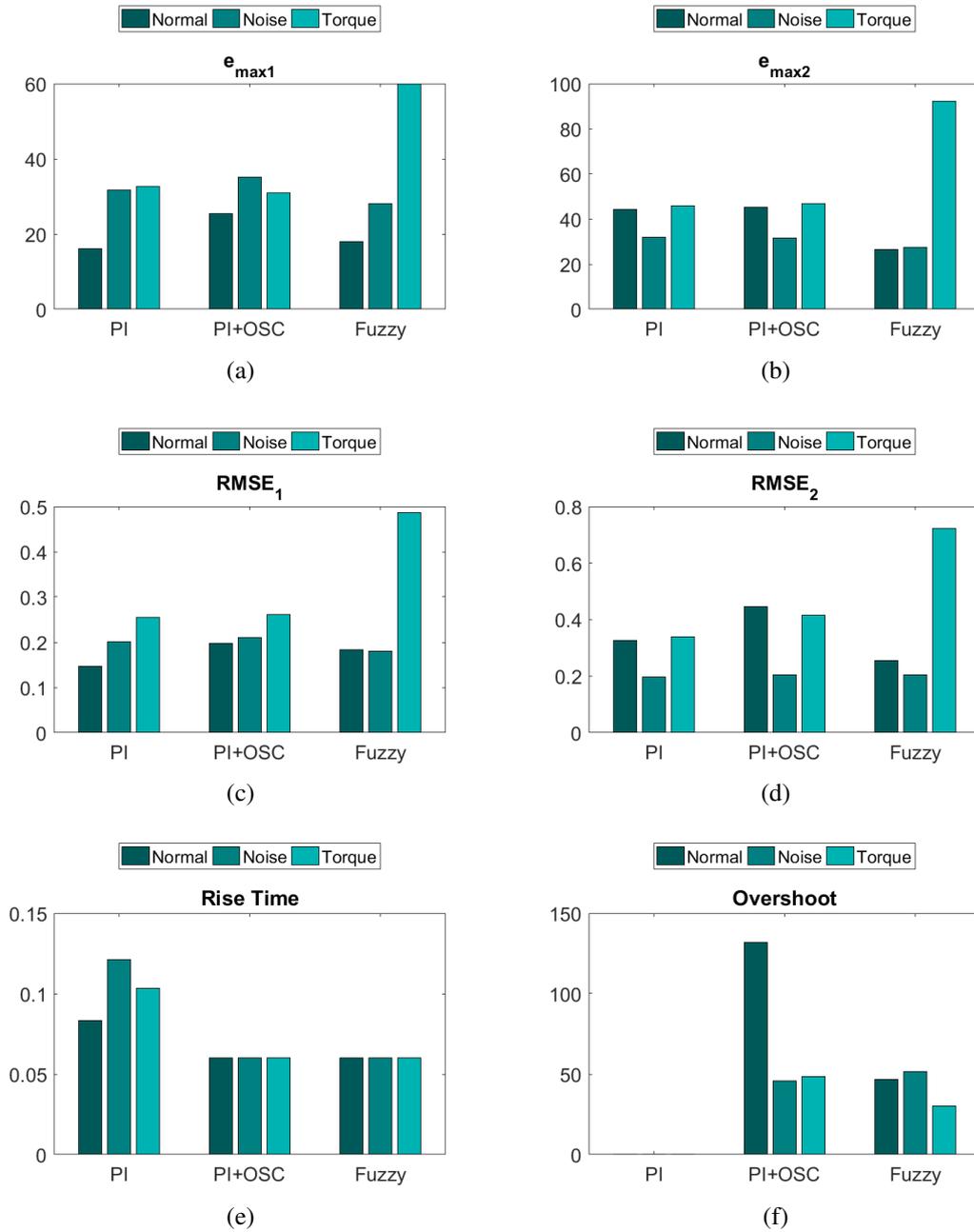


Figure 5.35: Bar charts for the model which is 1.70 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

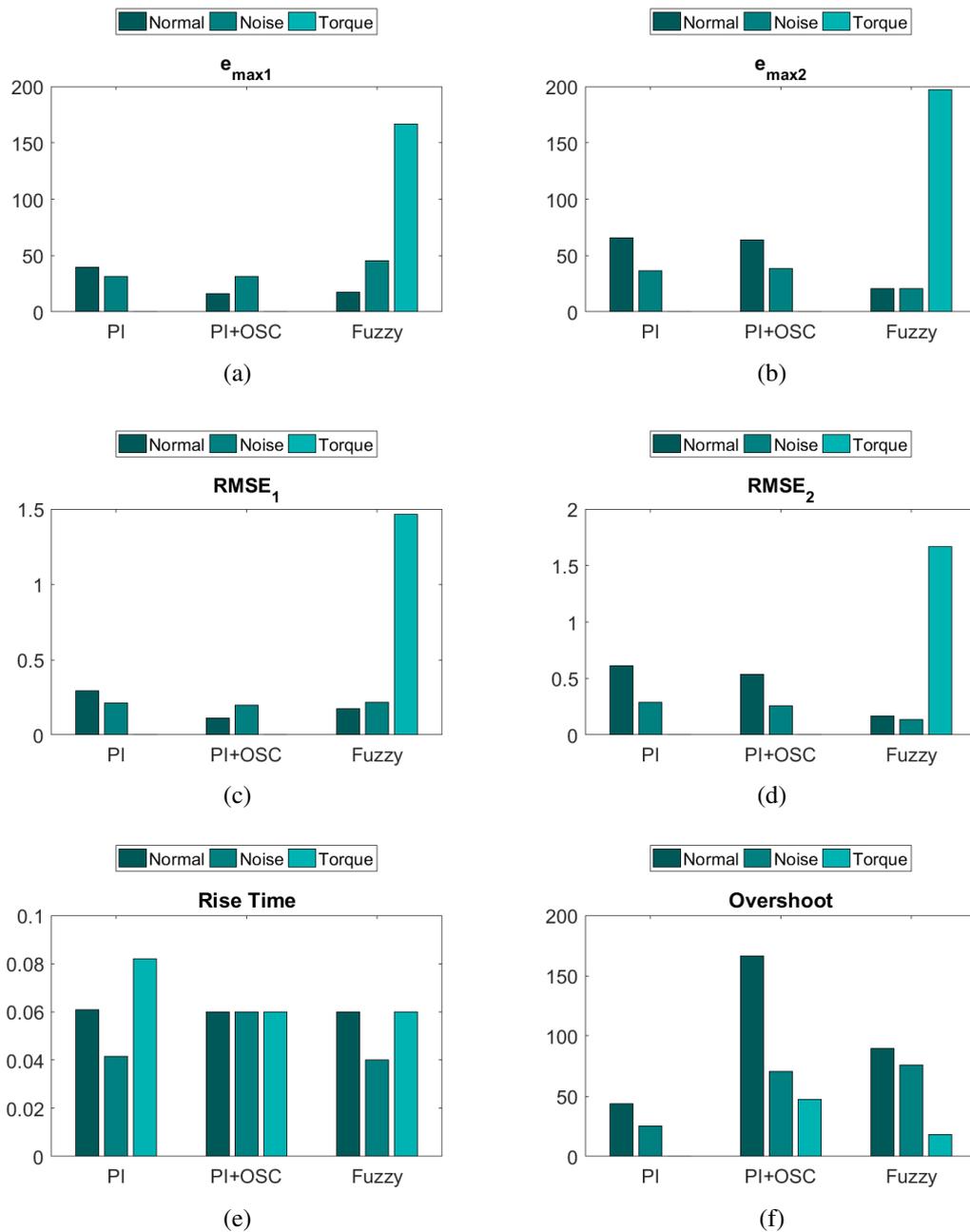


Figure 5.36: Bar charts for the model which is 1.90 m tall. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.3.7 Study about the influence of fatigue

5.3.7.1 Method

OpenSim standard muscle-tendon model does not simulate fatigue (Delp et al., 1990). Then, we added fatigue to the model and tested different time constants. FES sections with participants with SCI last a few minutes to an hour (Hunt et al., 2002), while computer simulations last a few seconds (Abdulla et al., 2014). Hence, time constants have to be small to see the fatigue effect during simulations. We tested two values for the resting time constant (T_r): 30 s and 60 s. The fatigue time constant (T_f) was 30 s in both cases. These values produce a fatigue factor smaller than 0.5 (see an example in Figure 4.3). The fatigue effect was based on the equations of the basic framework in Chapter 4.

5.3.7.2 Results and analysis

Simulation results are in Figures 5.37 and 5.38. It is possible to see that the maximum error for QH-Manual visibly decreased when compared with the simulation without environment modifications. The bar charts of Figures 5.12 and 5.41 show that the maximum error decreased to approximately half the value. Also, controllers which contain a PI component had a longer rise time at 20s, i.e., during the transition from 200°/s to 300°/s (see Figures 5.37 and 5.38). QHG-PI also had an error that decreased over time while the fatigue was increasing.

The average activation for simulations with fatigue is shown in Figures 5.39 and 5.40. In general, the average activation did not change much for $T_r = 30$ s and was slightly greater for $T_r = 60$ s. For example, the average activations of QHG-Fuzzy were the same for $T_r = 30$ s and without environment modifications. It is important to note that the control signal is multiplied by $(1 - f)$, where f is the fatigue factor. Hence, the controller had to increase the signal to overcome the fatigue. However, the increase for $T_r = 30$ s was not enough to influence the control, and the influence for $T_r = 60$ s was small.

The outcome of the addition of fatigue can be seen in the bar charts of Figures 5.41 and 5.42. QH-PI 2 had the lowest maximum errors and RMSEs between the controllers without gluteus muscle group activation, except the maximum errors in t_2 . The angular velocity of QHG-PID for $T_r = 30$ s had a visible instability, and this instability influenced the performance. On the other hand, QHG-PID had the smallest maximum errors and RMSEs for $T_r = 60$ s. The maximum errors and RMSEs of QHG-Fuzzy were slightly bigger than the results for QHG-PI and QHG-PI+OSC.

The rise time of the controllers without gluteus muscle group activation was 0.04 s in simulations without environment modifications and increased to 0.06 s in simulations with fatigue. The rise time also increased for controllers with gluteus muscle group activation, except for QHG-PI+OSC. The overshoot of QH-Manual decreased to below 60°/s, and it decreased to below 80°/s for QH-PI 1. The overshoot of QH-PI 2, which bellows 50°/s in simulations without environment modifications, increased to above 70°/s. QHG-PI and QHG-PID did not have overshoot, and the value for QHG-PI+OSC and QHG-Fuzzy decreased when compared with simulations without environment modifications. As in previous simulations, QHG-PI+OSC would not have overshoot if we consider the output of the controller oscillator instead of the desired velocity. Also, it is important to note that QHG-PI+OSC could couple with the ideal controller or follow it with a constant difference phase.

When the muscles are stimulated by FES, the fatigue effects take some minutes to appear. If we take this into account, simulations with fatigue would have the same overshoot and rise time of simulations without environmental modifications.

Even though the average activation of QH-PI 2 was three times bigger than the average activation of QH-PI 1, QH-PI 2 had a better performance. Hence, the addition of the opposing muscle activation and the parameters

for enabling the controller most of the time granted a better performance in simulations with little fatigue. The performance of controllers with the gluteus muscle group was similar, Except for simulations in which an instability appeared in the angular velocity, which may be caused by model limitations. However, in general, QHG-PI was more robust.

The influence of the fatigue in the control signal was small. If we had used a bigger fatigue factor, muscle activation could have increased until the maximum value (1.0). In future works, we plan to find the rest time constant which causes the muscle activation to be equal to the maximum value. As the real fatigue increases over time, a controller which is robust with bigger fatigue effects can be used for more extended tests.

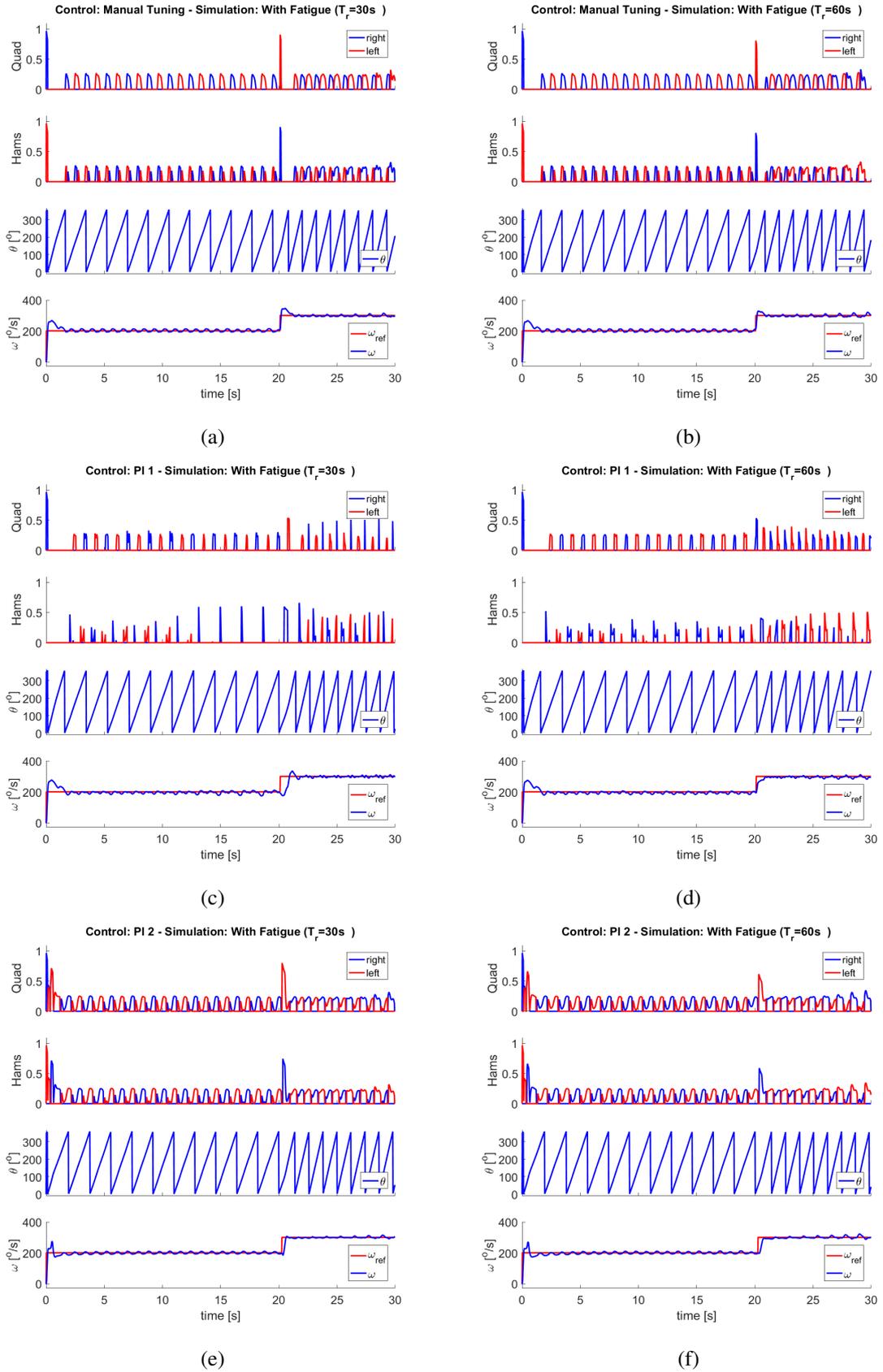


Figure 5.37: FES cycling simulation results of the controllers activating quadriceps and hamstrings. The controllers are (a, b) manually tuned, (c, d) the previous PI controller tuned by GA (PI 1) and (e, f) the new one (PI 2). We performed simulations with fatigue and (a, c, e) $T_r = 30$ s and (b, d, f) $T_r = 60$ s.

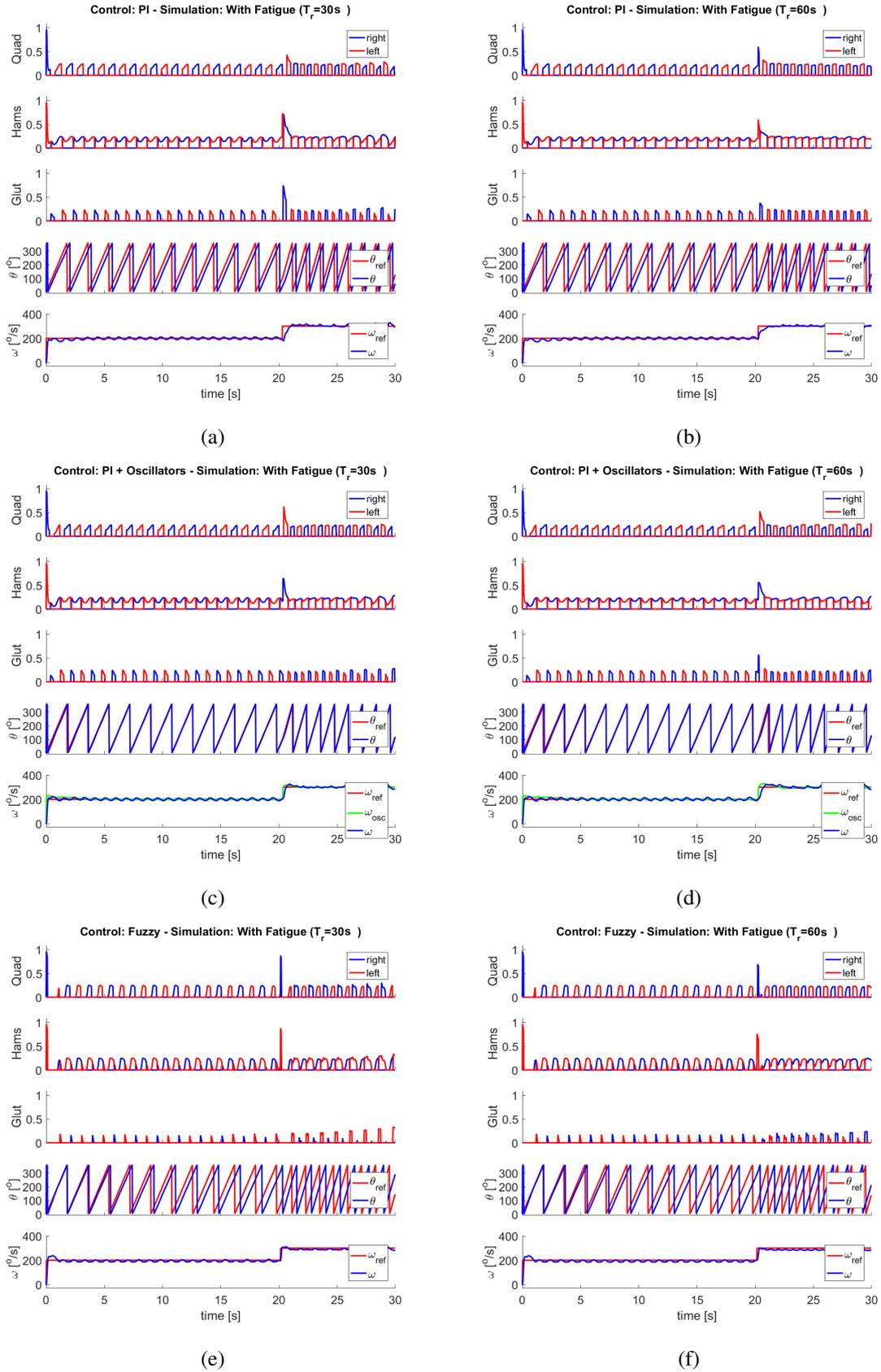
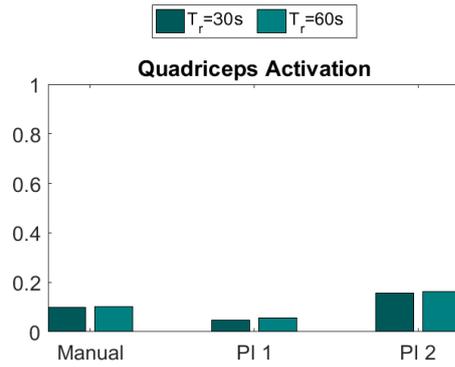
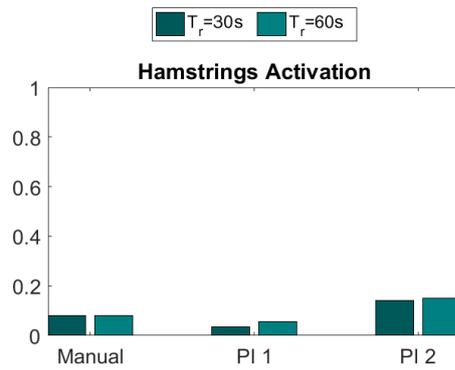


Figure 5.38: FES cycling simulation results of the controllers activating quadriceps, hamstrings and gluteus. The controllers are (a, c, e) the PI controller and (b, d, f) the PI controller with coupled oscillators. The controllers are (a, b) the PI controller, (c, d) the PI controller with coupled oscillators and (e, f) the FLC (Fuzzy). We performed simulations with fatigue and (a, c, e) $T_r = 30$ s and (b, d, f) $T_r = 60$ s.

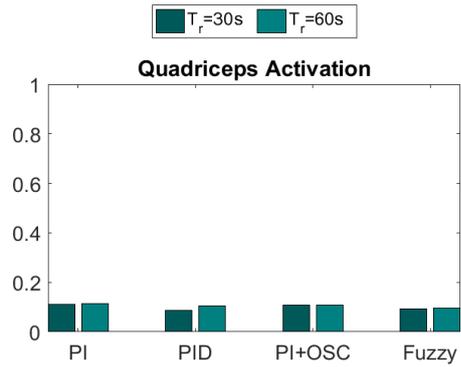


(a)

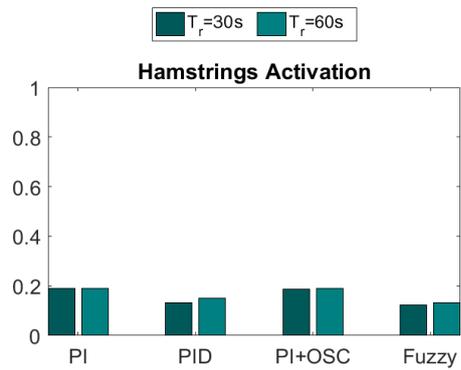


(b)

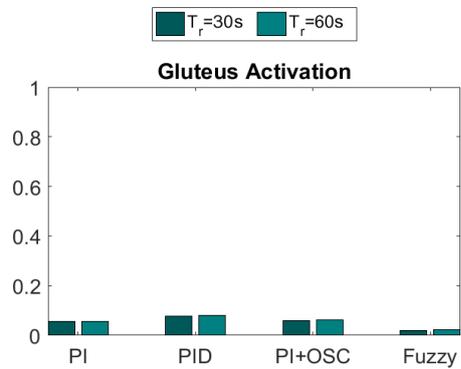
Figure 5.39: Bar charts of the controllers activating quadriceps and hamstrings. The controllers are manually tuned, the previous PI controller tuned by GA (PI 1) and the new one (PI 2). (a) is the average quadriceps activation, and (b) is the average hamstrings activation.



(a)



(b)



(c)

Figure 5.40: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). (a) is the average quadriceps activation, (b) is the average hamstrings activation, and (c) is the average gluteus activation.

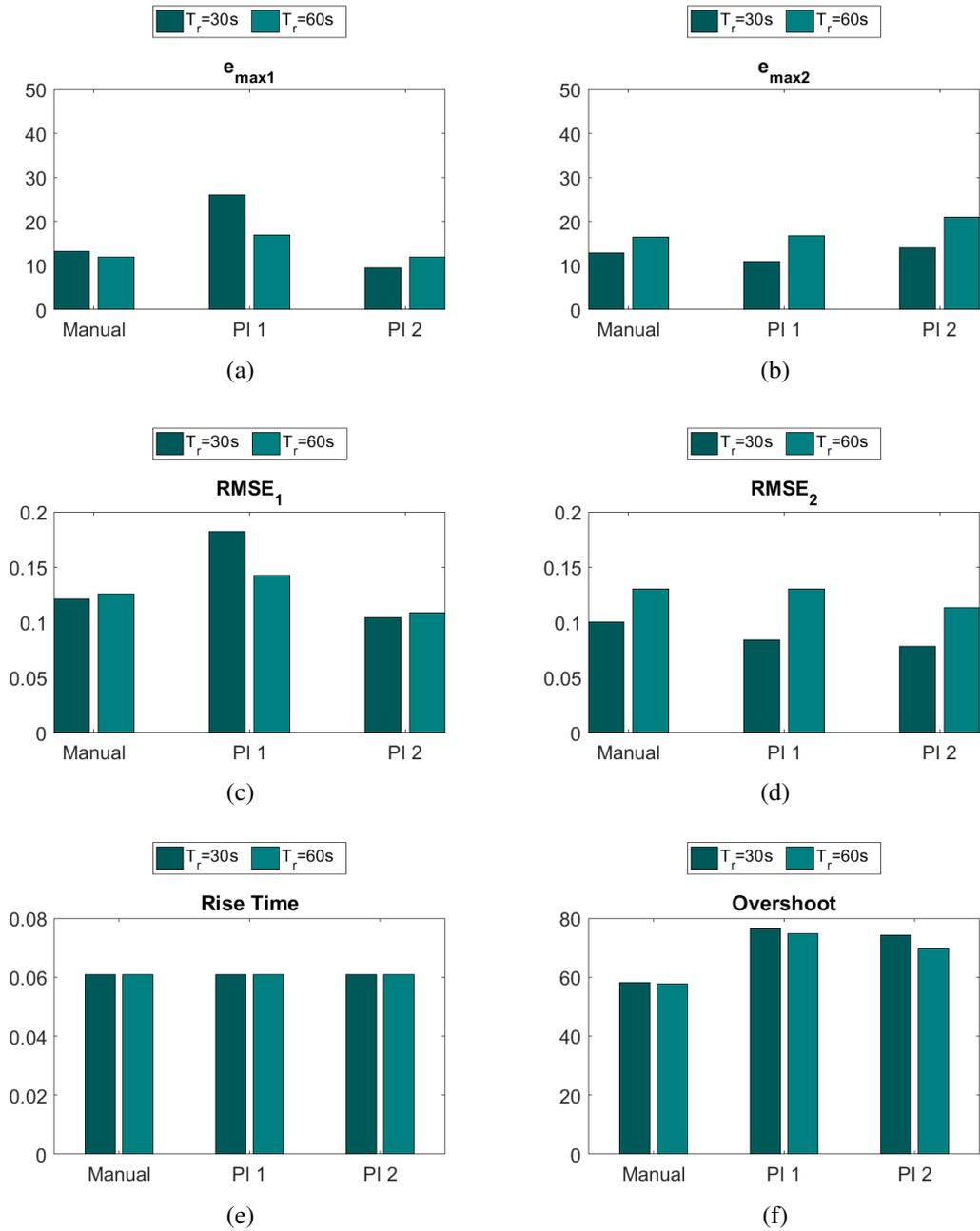


Figure 5.41: Bar charts of the controllers activating quadriceps and hamstrings. The controllers are manually tuned, the previous PI controller tuned by GA (PI 1) and the new one (PI 2). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

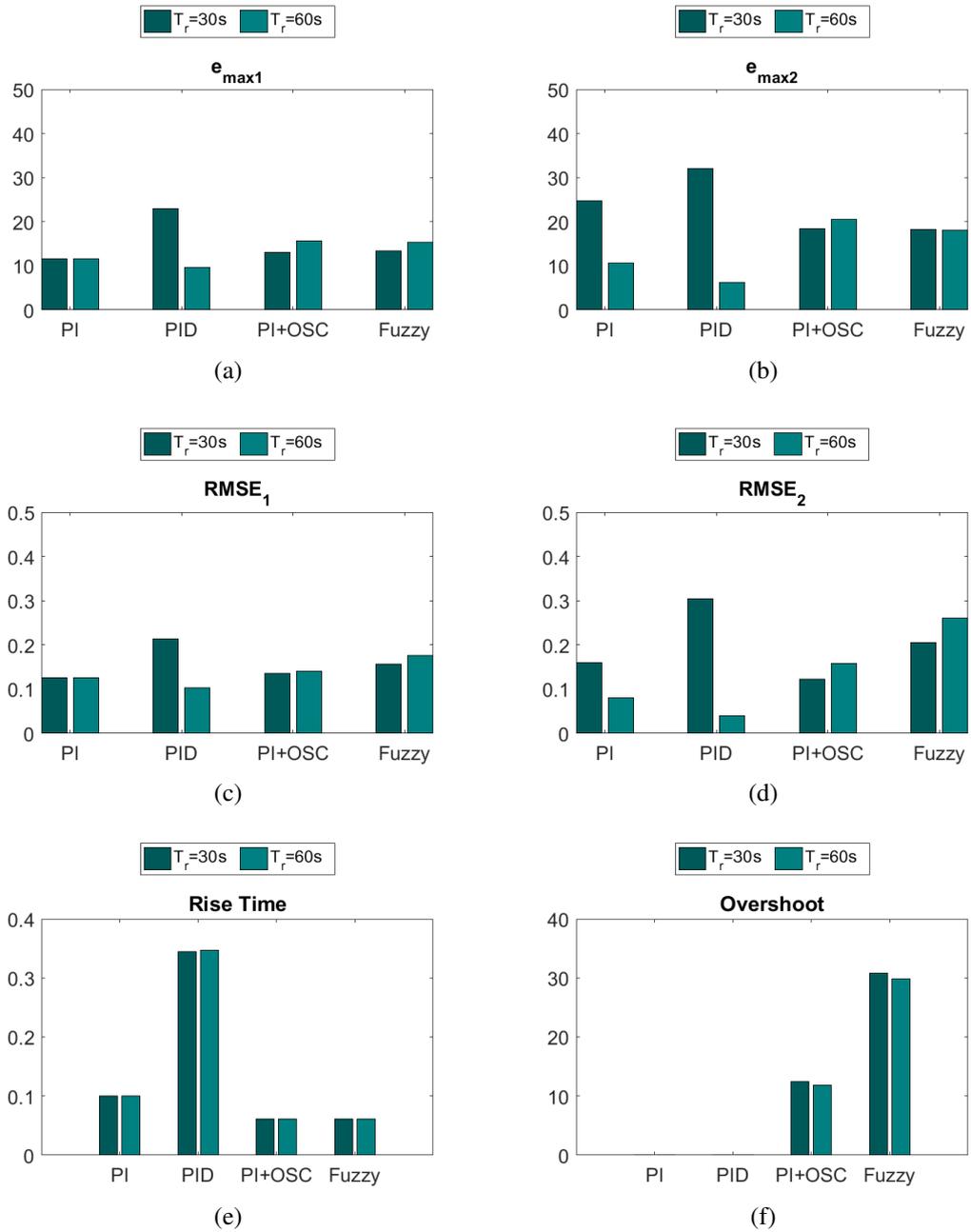


Figure 5.42: Bar charts of the controllers activating quadriceps, hamstrings and gluteus. The controllers are a PI, PI with coupled oscillators (PI+OSC) and FLC (Fuzzy). The measures are maximum error for (a) t_1 and (b) t_2 , RMSE for (c) t_1 and (d) t_2 , (e) rise time, and (f) overshoot.

5.4 MUSCLE WORK RATE AND ENERGY COST EFFICIENCY

Considering one of the major benefits of FES-assisted physical exercise involves cardiovascular health, we decided to use muscle work rate and energy cost efficiency to compare FES applications since both cycling and rowing produce mechanical work and the movements have a metabolic energy cost. The measures for QHG-PI, QHG-Fuzzy and QHG-PI+OSC during t_1 are shown in Figures 5.44. We did not use QHG-PID because it did not have the activation of opposing muscles. The muscle energy expenditure was calculated from simulations without environment modifications. As the musculoskeletal models used for FES cycling and FES rowing have different numbers of muscles, we used only the muscles used by the FSC.

QHG-PI+OSC had a slight instability during simulation, and it affected the energy cost. However, the instability did not pass on to the energy cost efficiency (see Figure 5.44.b). Note that the work rate has intervals of 0 W/kg since the negative work was not calculated. These intervals are small for QHG-PI+OSC when compared with QHG-PI and QHG-Fuzzy. Most of the time the energy cost is below 50 W/kg, and the work rate is below 30 W/kg.

The average energy cost efficiency for FES cycling controllers is shown in Figure 5.43. QHG-PI+OSC had the highest efficiency, which was 26.51%. Additionally, QHG-PI+OSC had an average activation similar to QHG-PI and performance measures equal or better than QHG-PI. However, the sharp slopes in Figure 5.44.b imply that QHG-PI+OSC requires more power from the muscles.

Hunt et al. (2012) have found a total work efficiency of 29.80% for a neurologically intact male subject aged 19 with body mass 75 kg during volitional cycling. The total work efficiency for FES cycling with the same participant was 12.50%. The results we found are similar to the volitional cycling. However, the efficiency is measured for different functions of the body, so it is not possible to make a precise comparison.

In summary, the addition of coupled oscillators improved the energy cost efficiency of FES cycling simulations. If we include the general performance of QHG-PI+OSC, it proved to be a useful alternative method to improve real tests.

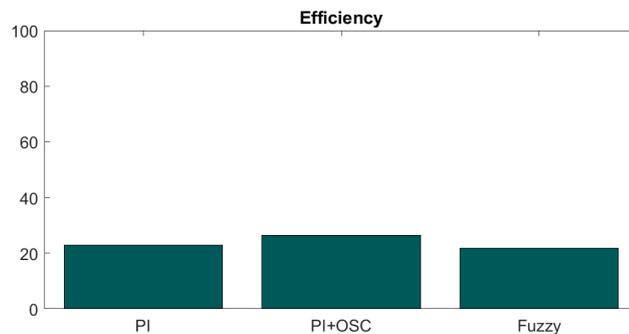
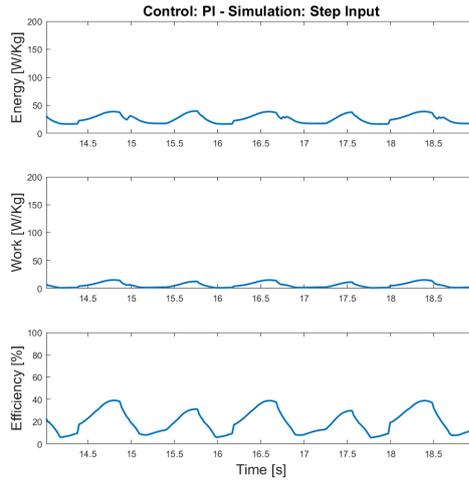
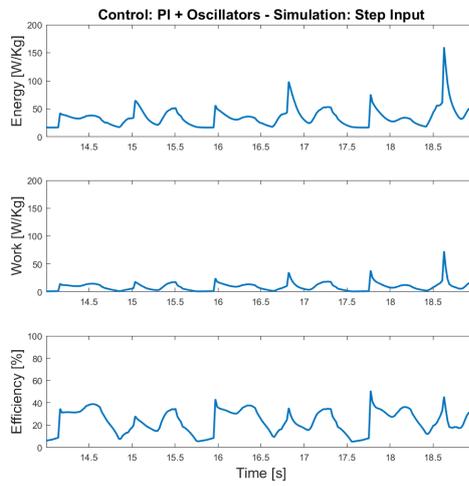


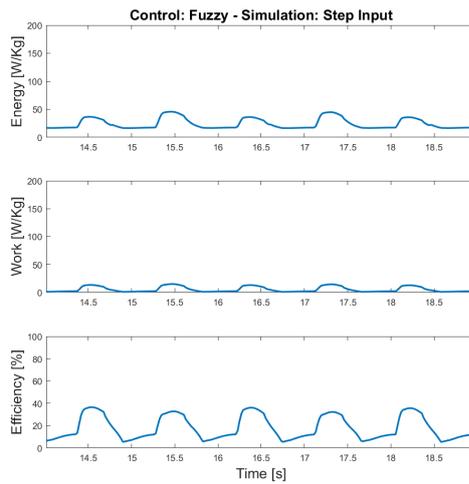
Figure 5.43: Energy cost efficiency of FES cycling with (a) PI controller, (b) PI controller with coupled oscillators (PI+OSC), and (c) FLC (Fuzzy). The average energy cost efficiency was defined as the average total work over the average total metabolic energy cost (Section 4.5).



(a)



(b)



(c)

Figure 5.44: Metabolic energy cost, mechanical work rate and energy cost efficiency of FES cycling with (a) PI controller, (b) PI controller with coupled oscillators, and (c) FLC (Fuzzy). The energy is expressed in W/kg of muscle tissue.

6

FES ROWING

This chapter presents the development of a musculoskeletal model and control strategies for FES rowing. Different from FES cycling, it consists of a total body exercise. The rowing movement has a specific pattern, which features alternating contraction of extensor and flexor muscles of the legs in a rhythmic pattern while the arms pull a handle.

First, the musculoskeletal model is presented. We used a standard model from OpenSim and added new objects and properties using Blender and Matlab. We used the measures of a real rowing ergometer (see Figure 6.1) to develop the additional parts of the model. Then, the next section presents a series of simulations involving different control strategies. Among the evaluated control strategies, there is an automatic control with different constant activation levels. Also, we explored the periodicity of the rowing movement to develop a controller based on attractor dynamics. The chapter concludes presenting the results of energy expenditure for FES rowing.

6.1 MODEL

6.1.1 Initial Model

We used the 3D Gait Model with Simple Arms from Hamner et al. (2010) as a starting point for the FES rowing model. It was chosen because its simplicity allows fast simulations. It features head-trunk, arms, pelvis, and legs segments.

Nevertheless, the model muscles from the arms are not included. Furthermore, only the muscles Biceps Femoris Long Head (*bifemlh*), Biceps Femoris Short Head (*bifemsh*), Rectus Femoris (*rect_fem*) and Vastus Intermedius (*vast_int*) were enabled to reduce the calculations of the simulation. The quadriceps group consists of Rectus Femoris and Vastus Intermedius, while the hamstrings group consists of Biceps Femoris Long Head and Biceps Femoris Short Head.

The degrees of freedom of the model include the movement of hip (*hip_flexion*, *hip_adduction*, *hip_rotation*), knee (*knee_angle*), ankle (*ankle_angle*), subtalar (*subtalar_angle*), metatarsophalangeal (*mtp_angle*), lumbar (*lumbar_extension*, *lumbar_bending*, *lumbar_rotation*), acromial (*arm_flex*, *arm_add*, *arm_rot*), elbow (*elbow_flex*), radio-ulnar (*pro_sup*) and wrist (*wrist_flex*, *wrist_dev*). The names in parenthesis are the equivalent



Figure 6.1: Ergometer for rowing. We used the measures of this ergometer to develop the additional parts of the FES rowing model.

objects in the musculoskeletal model file.

The original model was meant to perform walking simulations, so the arms do not have proper limit angles for rowing. The original radio-ulnar pronation and supination (*pro_sup*) of the model feature maximum limit angle of 90° . This limit was not enough to enable grabbing the handle. Also, the limit of on humans is approximately 180° . Hence, we changed the maximum limit angle to 180° .

6.1.2 Added objects

To simulate the rowing movement, we added three objects to the model: an ergometer base, a seat, and a handle. The ergometer base is fixed on the ground. The seat is attached above the ergometer and may translate along the sagittal plane. The handle is attached at the end of the ergometer and may also translate along the sagittal plane. Handle and ergometer base are linked using a *Thelen2003Muscle*¹ to simulate the chain.

The three new objects have to be added to the kinematic chain of the model. The simplest method is to add the new objects to another kinematic chain, making the ergometer base a child of the ground, the pelvis a child of the seat, and the hands children of the handle. However, this method creates the problem of a loop in the kinematics, i.e., the pelvis and hands belong to two kinematic chains. OpenSim avoids this problem using Constraints². Constraints are objects that limit motion between two connected bodies. They are similar to joints, but it is not possible to add an actuator and then control constraints. In the case of the *PointConstraint* and *WeldConstraint*, the constraint rotates and is locked, respectively. It is important to note that a constraint automatically adjusts the angles and translations of the joints when the state of the model changes, which is similar to what inverse kinematics does. Constraints have a narrow margin of tolerance, and the forward dynamics will stop if the tolerance is not satisfied.

The new objects were added as follows:

- Ergometer Base as a child of ground with *WeldJoint*;
- Seat as a child of Ergometer Base with *SliderJoint*;
- Handle as a child of Ergometer Base with *SliderJoint*;
- Pelvis as a child of Ergometer Base with *WeldJoint*;
- Foot (*calcn*) connected to Ergometer Base with *WeldConstraint*;
- Hand connected to Handle with *PointConstraint*.

The real rowing ergometer also has a slight inclination of 1° in the sagittal plane. We added this inclination in the joint that connects the ergometer base and the ground.

6.1.3 Rowing pose

The simulation of a model for FES rowing was used to create new controllers for the real system. Hence, the pose of the model had to resemble a person using the equipment (see Figure 6.1). The model automatically adjusted its pose to the ergometer. However, some joints have to be locked to prevent odd movements. Also, others degrees of freedom were locked to represent the sitting pose and ergometer geometry. The initial angles and which joints are locked are shown in Table 6.1. Figure 6.2 shows the musculoskeletal model in the rowing pose. It is possible to simulate other ergometers changing the parameters of Table 6.1.

¹https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1Thelen2003Muscle.html

²<https://simtk-confluence.stanford.edu/display/OpenSim/OpenSim+Models#OpenSimModels-KinematicConstraintsinOpenSim>

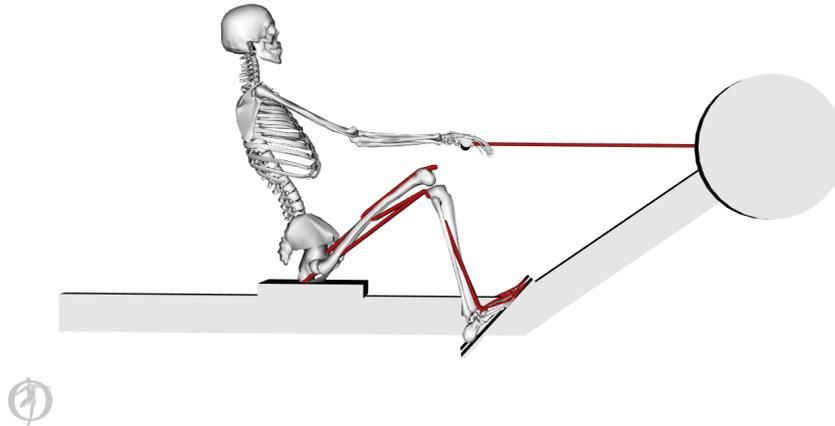


Figure 6.2: Rowing pose. The pose of the model had to be similar to the pose in the real system. Head-trunk and pelvis were fixed to simulate a person on a sitting position.

6.1.4 Motion restriction and knee reaction force

One limitation of OpenSim is that it does not respect the joint limits during Forward Dynamics. For example, if the minimum limit of *knee_angle_r* is -120° , it is possible to exceed this value, which breaks the constraints and the rowing movement. The movement has to stop when the legs achieve maximum extension and flexion. We used *CoordinateLimitForce*³ to apply limits to the ankle angle. What *CoordinateLimitForce* does is create a force that opposes the movement, so the body stops before it exceeds the limit.

CoordinateLimitForce can also be used for two different goals: it can simulate a safety stop, and it is possible to use it to calculate joint reaction force. Safety stops were used to limit the seat motion and prevent hyperextension and hyperflexion in other works (Davoodi et al., 2002; Hussain et al., 2008). Also, when the seat stops at a safety stop and the muscle groups are still applying force, the joints can be damaged because of the load. As the *CoordinateLimitForce* applies a force to halt the motion, we can use it to have an idea of the load on the knees.

We reduced the limits of the knee joint to -30° and -115° , while the originals are 10° and -120° . These limits restrict the motion of the seat to a range of 0.36 m from one limit to the other. In the tests of Davoodi et al. (2002), the distance that the seat travels is about 0.48 m for a participant without paraplegia and 0.29 m for the participants with SCI. The distance is smaller for participants with SCI mainly because of the safety stops.

6.1.5 Voluntary upper body movement

Note that the arms of the musculoskeletal model (see Figure 6.2) do not include any muscles. Also, the chain which connects the handle and the ergometer base restricts the handle movement to the translation along the sagittal plane. The purpose of this FES rowing model is to create and test controllers for people with paraplegia, i.e., only the muscles of the legs are activated. Hence, the movement of the arms is used just to provide feedback from the handle. Thus, we used a force actuator in the hands that activates when the model has to simulate the handle pulling. When there is no force applied, the muscle-tendon that represents the chain tends to pull the arms. This muscle-tendon is always active with an intensity of 0.1. Otherwise, it would relax and lose the pulling effect due to its passive elastic properties. The optimal fiber length is 1.01 m, and the resisting length of the tendon is 0.30 m.

³https://simtk.org/api_docs/opensim/api_docs/classOpenSim_1_1CoordinateLimitForce.html

Table 6.1: Initial angles and which joints are locked in the FES rowing model. The joint names follow the nomenclature from OpenSim standard. `cylinderToBase` is the joint that connects the handle to the ergometer base.

Joint	Initial Angle (°)	Locked
hip_flexion_r	114.218	no
hip_adduction_r	-0.548	no
hip_rotation_r	1.551	no
knee_angle_r	-113.461	no
ankle_angle_r	31.619	no
subtalar_angle_r	7.857	no
mtp_angle_r	0.000	yes
arm_flex_r	50.631	no
arm_add_r	-0.684	no
arm_rot_r	-2.459	yes
elbow_flex_r	27.330	no
pro_sup_r	180.000	yes
wrist_flex_r	-6.156	no
wrist_dev_r	-0.129	yes
lumbar_extension	0.000	yes
lumbar_bending	0.000	yes
lumbar_rotation	0.000	yes

Joint	Initial Angle (°)	Locked
hip_flexion_l	114.218	no
hip_adduction_l	-0.548	no
hip_rotation_l	1.551	no
knee_angle_l	-113.461	no
ankle_angle_l	31.619	no
subtalar_angle_l	7.857	no
mtp_angle_l	0.000	yes
arm_flex_l	50.631	no
arm_add_l	-0.684	no
arm_rot_l	-2.459	yes
elbow_flex_l	27.330	no
pro_sup_l	180.000	yes
wrist_flex_l	-6.156	no
wrist_dev_l	-0.129	yes
seatToBase	0.272	no
cylinderToBase	0.226	no

It was not possible to lock the rotation of the hands to the handle because of the tolerance of the constraints. The forward dynamics stopped if the tolerance was not satisfied. As there was no actuator controlling the arms, we connected then hands to the handle using *PointConstraints* and increased the inertia of the hands to reduce rotational movements. The new inertia is 1000 times bigger than the original one.

6.2 NUMERICAL INVESTIGATION

The next subsections present the results acquired from FES rowing simulations. They were divided according to the type of control strategy. We used the same prerequisites to acquire the data.

All simulations feature 10 s duration and sampling time of 0.02 s. The FSC for FES rowing was based on Davoodi et al. (2002). The states of the motion cycle are shown in Figure 3.2. First, the participant assumes the Catch position. Hamstrings are activated to prevent the model from getting out this position. Then, the control enters the Drive state in which the quadriceps are activated until the Handle Pull position. In the Handle Pull position, the quadriceps continue activated while the model pulls the handle. After finishing the pulling, the control enters the Recovery state in which the hamstrings are activated, and the model goes back to the Catch position. We used the handle position to know when the model has to start and finish the handle pulling. The FSC knows that the model is in the Catch position by the seat position.

We used GA to find better parameters and compare the performance of new strategies. The GA parameters were obtained following the framework described in Section 4.4, and the fitness function tries to maximize the velocity. The fitness function is the combination of:

- Sum of the absolute value of the seat velocity;

- Sum of the absolute value of the derivative of the angle of the motion.

Since the rowing movement is rhythmic, it is possible to represent the motion as a oscillator. The angle of the motion is 0° in the Catch position, and 180° in the Handle Pull position. This angles goes from 0° to 180° when the model is in the Drive state, and it goes from 180° to 360° when the model is in the Recovery state. The angle of the motion is calculated as

$$s_n = \frac{s + 0.07}{0.29 + 0.07} - \frac{0.29 - s}{0.29 + 0.07}, \quad (6.1)$$

$$\theta = \text{atan2}(-v, s_n), \quad (6.2)$$

where s is the seat position, v is the seat velocity, and θ is the angle of the motion. θ may be used to couple the model with another oscillator. Note that the term s_n is not the term s normalized. When the model is in the Catch position ($s = 0.29$ m), $s_n = 1$, and when the model is in the Handle Pull position ($s = -0.07$ m), $s_n = -1$. That means s_n acts like $\cos(\theta)$. v did not pass through the same process because we do not know the maximum and minimum velocity previously.

Note that it is only possible to calculate a angle from 0° to 180° when only the seat position is used. The seat velocity is used because it is negative during Drive state (0° to 180°) and positive during Recovery state (180° to 360°), so we can use the operator atan2 to calculate the angle from 0° to 360° .

An example of simulation result with the data of the seat is shown in Figure 6.3.a. As the activation is the same for both legs, the lines in the top two graphs are superimposed. If the value is 0, the muscle is relaxed, and if the value is 1, it applies the maximum isometric force. The angle of the third graph is important to observe the coupling ability of oscillators. An example of simulation result with the data of the handle is shown in Figure 6.3.b. Control strategies used are described in the following subsections.

Additionally, we used the simulation data to calculate maximum velocity, the number of strokes per minute and average muscle activation. As simulations had only 10 s, the number of strokes per minute is the inverse of the period of a stroke multiplied by 60. There are two approaches to find the average muscle activation. The first is to use the mean equation

$$\bar{\gamma} = \frac{\sum_{i=1}^N \gamma[i]}{N}, \quad (6.3)$$

where $\bar{\gamma}$ is the mean, $\gamma[i]$ the muscle group activation at sample i and N the total number of samples. However, this approach takes into account the samples when the activation is 0, which decreases the mean. Another alternative is to divide the sum of activations only per the number of samples with nonzero activation. For example, for a constant quadriceps activation of 0.50 during a quarter of a cycle, the average for nonzero activation is 0.50, while the total average is 0.13. The average for nonzero activation is equal to the constant activation because it contemplates only a quarter of samples. This method is the same used for FES cycling. An example of a bar chart of a performance measure is shown in Figure 6.4.

6.2.1 Automatic control with constant activation

The FSC was tested first with a constant activation signal. Experimental studies based on manual and automatic constant controllers used a stimulation intensity which produces maximum muscle force or sufficient to sustain a high velocity (Davoodi et al., 2002; Yusoff et al., 2014; Ektas et al., 2014). Simulation results for constant activations of 0.25, 0.50 and 0.75 are shown in Figure 6.6. It is possible to see that the controllers achieved the Handle Pull and Catch positions. The controller with constant activation of 0.25 achieved the Catch position after the Recovery state, but it started the Drive state soon after. Additionally, the model took a long time to exit the Handle Pull position during Recovery state with activation of 0.25. The other controllers

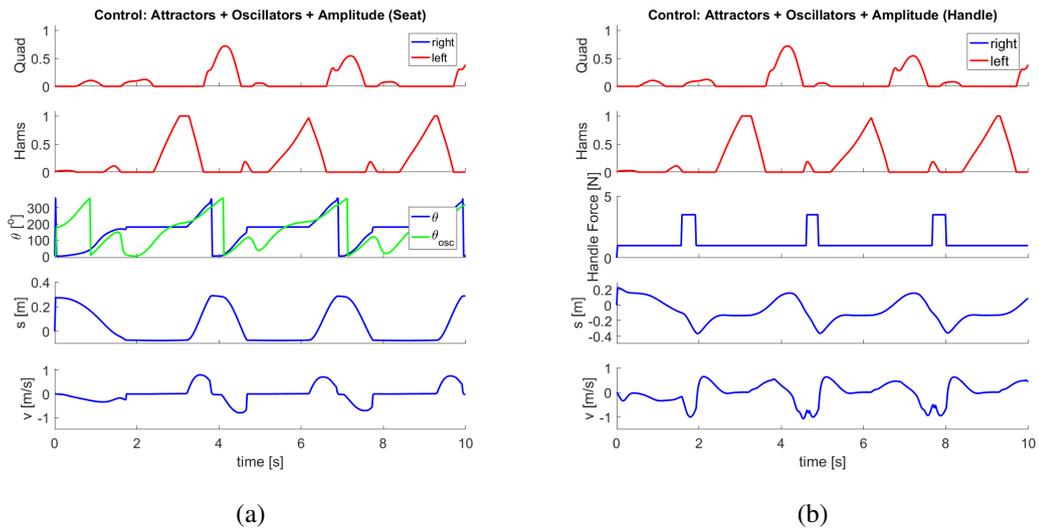


Figure 6.3: Example of a simulation result. The top two graphs are the activation of the muscle groups. In (a), the third graph in is the angle ($^{\circ}$) of the motion. 0° is the Catch position, and 180° is the Handle Pull position. The blue line is the angle acquired from the data of OpenSim, and the green line is the output of the coupled oscillators. The fourth graph is the position (m) of the seat, and the fifth graph is the velocity (m/s) of the seat. In (b) the third graph is the force (N) which simulates the handle pulling. The fourth graph is the position (m) of the handle, and the fifth graph is the velocity (m/s) of the handle.

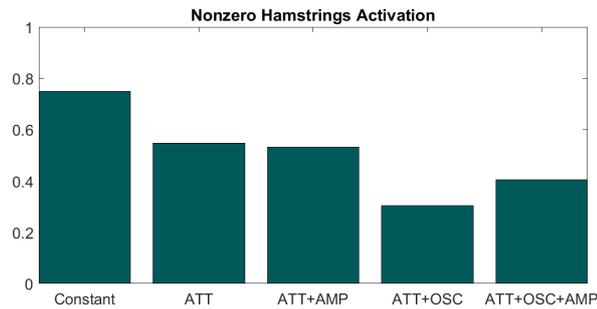


Figure 6.4: Example of a bar chart of a performance measure. Each bar is the performance of a control strategy.

also had a slow recovery compared to the Drive state, but the difference of duration was small. The handle pulling occurred when there was no muscle activation.

The performance measures are shown in the bar charts of Figure 6.5. The maximum velocity achieved by the controller with constant activation of 0.25 was 0.26 m/s, while the value for the controller with constant activation of 0.75 was 0.88 m/s. The number of strokes per minute also increased along with activation. The controller with constant activation of 0.75 achieved the highest value, which was 23.86 strokes/min.

The velocity increased with the increase of the activation level. The beginning of Recovery state was the most affected part of the motion cycle. It is possible to notice that activation level below 0.25 could not be enough to exit the Handle Pull position during Recovery state. Hence, controllers with constant activation need a high activation intensity to perform the Recovery state faster and achieve high velocities. However, high activation levels may cause fast fatigue effects, particularly in participants with SCI.

Simulations of Hussain et al. (2011) achieved approximately 20 strokes/min. They have developed a self-adaptive neuro-fuzzy control for FES rowing. In real tests, the system achieved approximately 24-28 strokes/min in Davoodi et al. (2002), and 16-17 strokes/min in Ektas et al. (2014) with an automatic control and constant

stimulation. Note that Ektas et al. (2014) have developed an alternative ergometer, consisting of a sliding foot stretcher and fixed seat system. Additionally, the seat traveled different distances in each study, and the handle pulling was counted as part of the motion cycle. Hence, even though the controller with constant activation of 0.75 had a high velocity, it is difficult to compare performances.

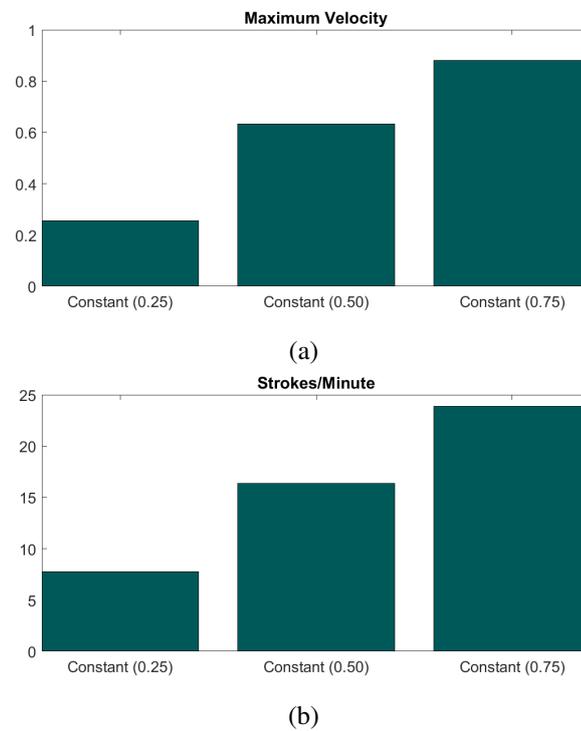


Figure 6.5: Bar charts for controllers with constant activation: (a) the maximum velocity (m/s) and (b) the number of strokes per minute.

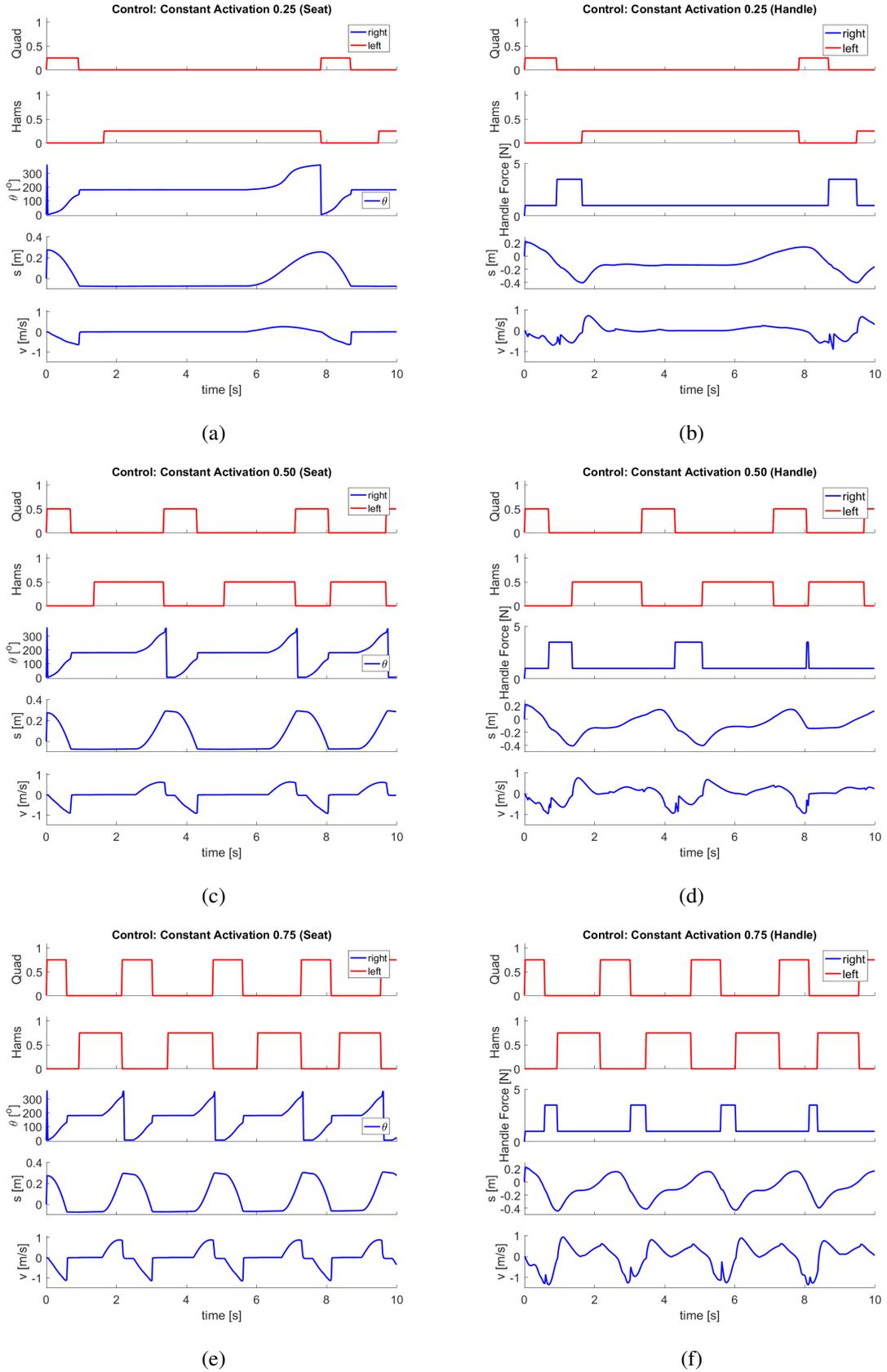


Figure 6.6: FES rowing simulation results for controller with constant activation of (a, b) 0.25, (c, d) 0.50 and (e, f) 0.75. The graphs show the measures of (a, c, e) the seat and (b, d, f) the handle.

6.2.2 Controller based on attractor dynamics

6.2.2.1 Method

Even though FES rowing has a rhythmic pattern, it cannot be classified as a pure cyclic movement. Only the muscles of the legs are assisted by FES, while the movement of the arms is voluntary. The FSC waits for the handle pulling to finish before it starts the Recovery state, so it is not possible to use a lower level controller for fully rhythmic movements.

Some studies have described Drive and Recovery states as different movements (Davoodi and Andrews, 2004; Hussain et al., 2010; Yusoff et al., 2014). The handle pulling was also done during Recovery state (Hussain et al., 2008). However, we can describe Drive and Recovery states as half cycles of an oscillation pattern. Additionally, the model has to stop in the Catch position and the Handle Pull position. Hence, the rowing motion can be divided into two discrete movements and one oscillation.

We implemented a controller that generates both discrete and the rhythmic movements for FES rowing based on the model of Schöner and Santos (2001). The layer capable of generating the output signal (timing layer) has three terms: two attractors consisting of the initial state and target state (i.e., Catch position and Handle Pull position), and a Hopf oscillator of amplitude 1 and frequency ω . The timing layer is described by Equations 2.22 and 2.23.

The original neural activity of the control layer, which decides the movement the system will do, has three equations corresponding to the normal form of a degenerate pitchfork bifurcation with an extra term to ensure that only one neuron is active. The layer starts with $\dot{u}_k = 0$, ($k = i, h, t$) and $[u_i u_h u_t] = [1 0 0]$. As a consequence, both sides of the equation

$$\alpha \dot{u}_i = \mu_i u_i - |\mu_i| u_i^3 - c(u_h^2 + u_t^2) u_i \quad (6.4)$$

are equal to 0, and all neurons become inactive. The neurons keep inactive even when the inputs of the control layer change.

Although there are solutions to this problem (Schöner and Santos, 2001), we decided to use the continuous-time, continuous variable neuron model of Matsuoka oscillators (Matsuoka, 1985) to implement the control layer. The new neural activity for the three primitives is shown in Table 6.2.

We used $S_h = 1$, $w = 1$, $t_r = 0.1$, $t_a = 20$ and $\beta = 0.009$ to simulate the same output of the original control layer. It is possible to control the rise time changing t_r . Note that the input S_h is constant, i.e., the Hopf neuron is always active. We can inhibit the oscillation and modify the movement to achieve one of the attractors alternating S_i and S_t between 0 and 1. S_i and S_t are equal to 1 when there is no movement and are set to 0 to activate the oscillation. An example of the activity of the control layer is shown in Figure 6.7, and an example of the outputs when the inputs change is shown in Figure 6.8.

Even though now the controller is the same for Drive and Recovery states, the inputs S_i and S_t still have to change if the model is trying to reach the Catch position or the Handle Pull position. The inputs to reach the

Table 6.2: Neural activity based on Matsuoka oscillators for the three primitives of the model based on attractor dynamics.

Initial neuron	Hopf neuron	Target neuron
$t_r \dot{x}_i = -x_i - \beta v_i + S_i$ $u_i = \max(0, x_i)$ $t_a \dot{v}_i = -v_i + u_i$	$t_r \dot{x}_h = -x_h - w u_i - w u_t - \beta v_h + S_h$ $u_h = \max(0, x_h)$ $t_a \dot{v}_h = -v_h + u_h$	$t_r \dot{x}_t = -x_t - \beta v_t + S_t$ $u_t = \max(0, x_t)$ $t_a \dot{v}_t = -v_t + u_t$

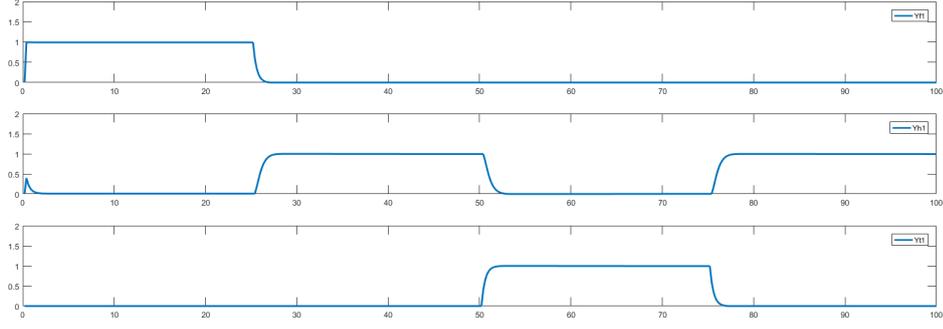


Figure 6.7: Neuronal activities u_i (up), u_h (middle) and u_t (down) of the control layer. Initial inputs are $[S_i, S_t] = [1, 0]$, then it changes to $[0, 0]$, $[0, 1]$ and $[0, 0]$ again at 25, 50, and 75 iterations respectively.

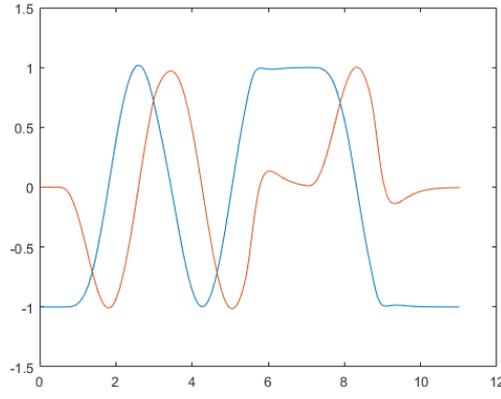


Figure 6.8: Outputs X (blue) and Y (red) of the timing layer. Initial inputs are $[S_i, S_t] = [1, 0]$, then it changes to $[0, 0]$, $[0, 1]$, $[0, 0]$ and $[1, 0]$ at 1.0, 5.0, 7.5 and 8.5 seconds respectively.

Handle Pull position are

$$S_i = 0, \quad (6.5)$$

$$S_t = 1 - (\tanh(10(7s_{hp} - s)) + 1)/2, \quad (6.6)$$

where s_{hp} is the Handle Pull position. The inputs to reach the Catch position are

$$S_i = 1 - (\tanh(10(x - 7s_c)) + 1)/2, \quad (6.7)$$

$$S_t = 0, \quad (6.8)$$

where s_c is the Catch position.

The inputs and outputs of the timing layer are the same (x and y). In FES rowing, the position of the seat (s) is the input, and the muscle activation (x) is the output, i.e., the input and output are not the same. A positive value of x activates the quadriceps, and a negative value activates the hamstrings. A system diagram for FES rowing with the timing layer and control layer is shown in Figure 6.9.

Additionally, we want a low activation for the attractors just not to exit the position while keeping a high activation during oscillation. Also, the original system has a constant frequency, so it needs feedback to adjust the frequency according to the actual motion. The new frequency of the Hopf oscillator is

$$\omega = (1 - q \cdot e \cdot x)\omega_{ref}, \quad (6.9)$$

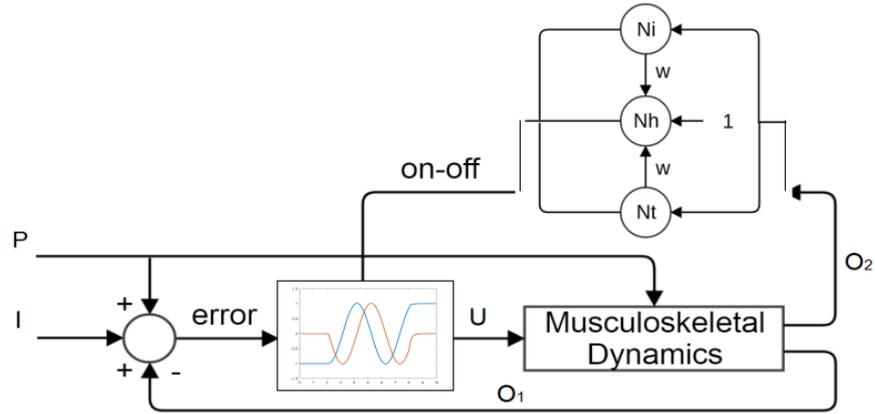


Figure 6.9: System diagram for FES rowing with the controller based on attractor dynamics. The control layer is equivalent to the FSC and receives besides other inputs the values of S_i and S_t . The output is the neural activity of the three neurons, which activate the primitives in the timing layer. The timing layer is equivalent to the lower level controller. The activation of the muscles (U) is given by x (blue line).

where ω_{ref} is the desired frequency, e is the difference between s and s_{hp} or s_c , and q is a constant gain. s_{hp} is used in the Drive state and s_c is used in the Recovery state. When the model is far from the Handle Pull position or Catch position, and the Hopf oscillator already achieved maximum amplitude, the feedback reduces the frequency of the oscillator to maintain a high amplitude until the model gets close to the desired position. We also changed the attractors of the initial state and target state to -0.1 and 0.1 to keep a low activation during the discrete movements. GA was used to find the parameters a , b , ω_{ref} and q . The values are shown in Table 6.3.

6.2.2.2 Results and analysis

Simulation results for the controller based on attractors dynamics (ATT) are shown in Figure 6.10. It is possible to see that ATT achieved the Handle Pull and Catch positions. In the first Drive state, the quadriceps activation is low because the seat got close to the Handle Pull position and S_t increased, i.e., the Hopf neuron was inhibited. In the next Drive states, the model took some time to exit the Catch position. Hence, the output x increased to a bigger activation, but the feedback weakened it after. The model also took some time to exit the Handle Pull position.

The bar charts from Figure 6.11 present the nonzero average activations for ATT and the controller with constant activation of 0.50. This constant activation was chosen because the velocity is similar to the controllers based on attractors dynamics. The nonzero average quadriceps activation of ATT was smaller than the value for the constant activation, mainly due to the initial Drive state and the low activation value during handle pulling. Figure 6.10 shows the quadriceps activation for ATT was bigger than 0.50 during the second Drive state. The nonzero average hamstrings activation of ATT was similar to the value for the constant activation, even though the maximum intensity was bigger than 0.50.

In the bar charts from Figures 6.5 and 6.12, it is possible to see that ATT achieved a maximum velocity bigger than the maximum velocity of the controller with constant activation of 0.50. Also, it was greater than the value for the controller with constant activation of 0.75. On the other hand, the number of strokes per minute of ATT was smaller when it is compared with the controller with constant activation of 0.50. Note that the model spent more time in the Catch position after Recovery state due to the remaining hamstrings activation (see Figure 6.10).

The control strategy based on attractor dynamics could complete the rowing motion, and it waited for the handle pulling to finish before it started the Recovery state (see Figure 6.10.b). Additionally, the controller achieved a bigger maximum velocity without increasing much the activation, although the number of strokes per minute decreased.

Table 6.3: The parameters for the controllers based on attractor dynamics. a is the gain of the attractors, b is the gain of the Hopf oscillator, q is the gain of the error, a_{c1} is the coupling constant, and ω_{ref} if the desired frequency of the oscillation. Attractor represents the controller based on attractors dynamics, and Attractors + Oscillators represents the controller based on attractors dynamics with the addition of coupled oscillators.

	Attractors	Attractors + Oscillators
a	8.32	8.32
b	1.34	1.34
q	2.32	-
a_{c1}	-	3.50
ω_{ref}	2.33	2.30

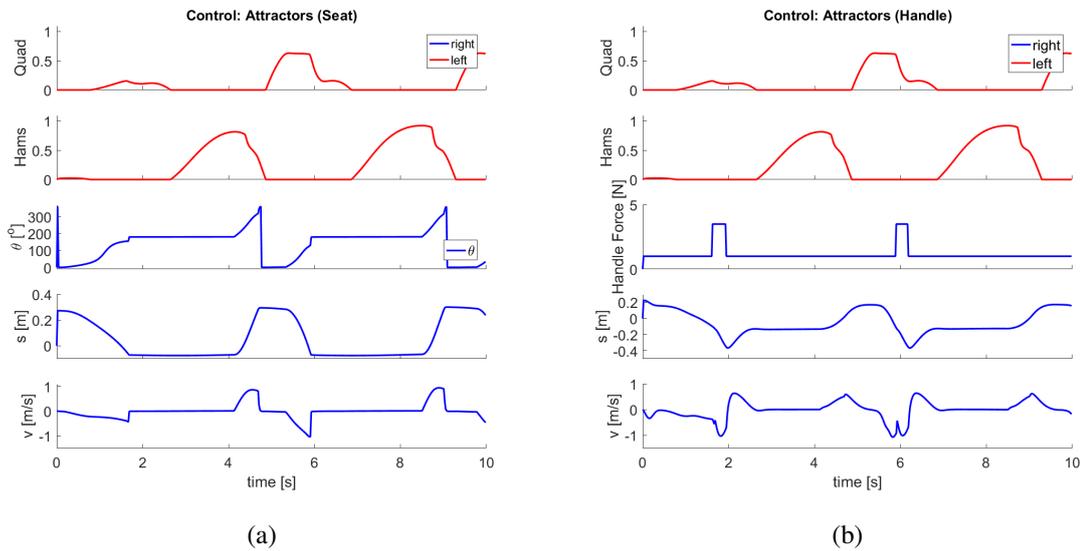
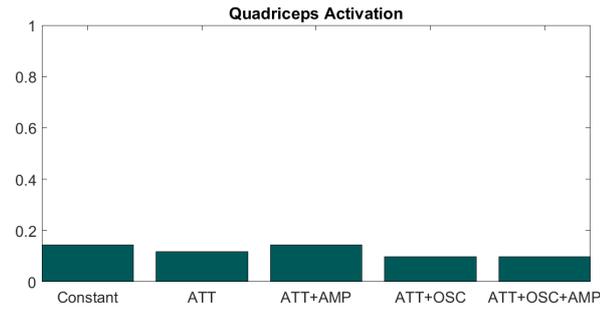
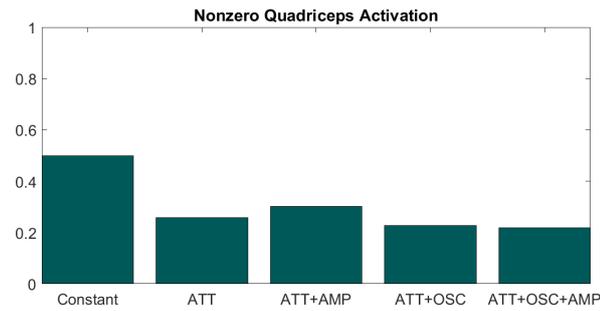


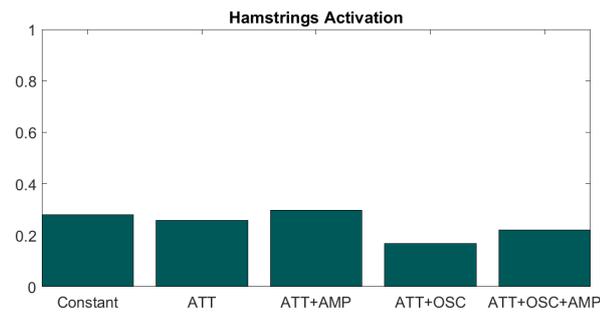
Figure 6.10: FES rowing simulation results for controller based on attractor dynamics. The graphs show the measures of (a) the seat and (b) the handle.



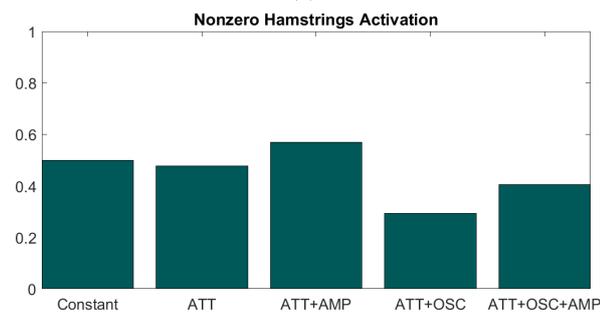
(a)



(b)



(c)



(d)

Figure 6.11: Bar charts for controllers with constant activation of 0.5 (Constant), based on attractor dynamics (ATT), and based on attractor dynamics with amplitude control (ATT+AMP), with coupled oscillators (ATT+OSC) and both (ATT+OSC+AMP): (a) average quadriceps activation, (b) nonzero average quadriceps activation, (c) average hamstrings activation, and (d) nonzero average hamstrings activation.

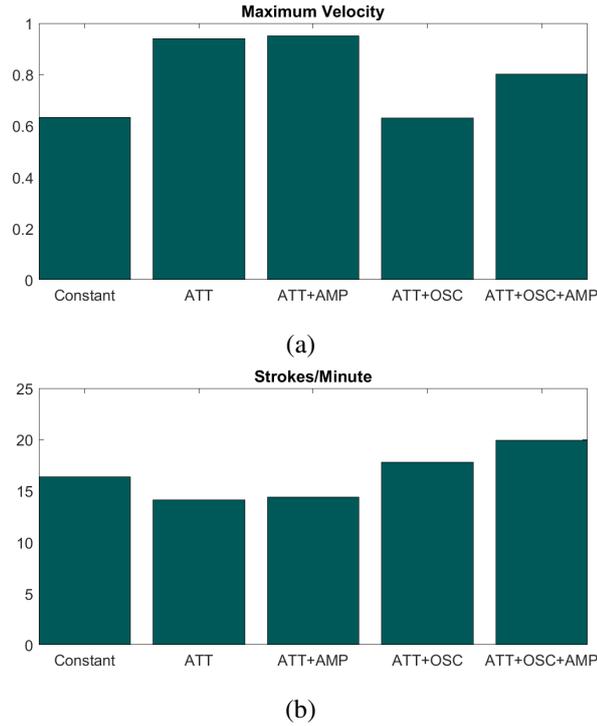


Figure 6.12: Bar charts for controllers with constant activation of 0.5 (Constant), based on attractor dynamics (ATT), and based on attractor dynamics with amplitude control (ATT+AMP), with coupled oscillators (ATT+OSC) and both (ATT+OSC+AMP): (a) the maximum velocity (m/s) and (b) the number of strokes per minute.

6.2.3 Amplitude control

6.2.3.1 Method

At least in one study, the participants preferred an automatic control system, but the cost regarding electrical stimulation was increased (Davoodi et al., 2002). Inspired by this report, a series of low-level controllers were developed to reduce the stimulation level while attempting to maintain other performance measures.

Since there is no feedback to control the amplitude of the Hopf oscillator, the output x always has to pass through 1 or -1 when the motion is longer than a half cycle (see Figure 6.8). x also represents the activation of the muscle groups. That means the activation has to pass through the maximum value too (see Figure 6.10). If the velocity of the seat is high enough and the seat is close to the Catch or Handle Pull position, the amplitude decreases but due to the small amplitude of the attractors (0.1). When the model is far from the Handle Pull position or Catch position, and the Hopf oscillator already achieved maximum amplitude, the feedback reduces the frequency of the oscillator to maintain a high amplitude until the model gets close to the desired position.

Davoodi et al. (2002) also stated that participants with SCI seem to release the handle much faster than the standard rowing technique in expectation that it will accelerate the slow beginning of the Recovery state. Due to this issue, we tried to control the amplitude of the Hopf oscillator directly. The new contribution of the Hopf oscillator to the output is

$$m = \max(1 - (\dot{\theta}_m - \omega_{ref}) \cdot |1 - x|, 0), \quad (6.10)$$

$$\dot{x}_h = |u_h|(b(m - r^2)x - \omega y), \quad (6.11)$$

$$\dot{y}_h = |u_h|(b(m - r^2)y + \omega x), \quad (6.12)$$

where $\dot{\theta}_m$ is the derivative of the angle of the motion, ω_{ref} is the desired frequency, and m is the amplitude.

The term $|1 - x|$ and the operator \max limit the amplitude to the interval $[0, 1]$. During the beginning of the Recovery state, the Hopf neuron is active, but the model takes some time to exit the Handle Pull position, i.e., $\dot{\theta}_m = 0$ rad/s. Then, the difference $(\dot{\theta}_m - \omega_{ref})$ increases m . If $\dot{\theta}_m$ is bigger than ω_{ref} during Drive and Recovery state, m decreases.

6.2.3.2 Results and analysis

Simulation results for the modified controller based on attractors dynamics (ATT+AMP) are shown in Figure 6.13. It is possible to see that hamstrings activation increased until the maximum level (1.0) at the beginning of the Recovery state, while it decreased during recovery. As a result, the time spent in the Handle Pull position decreased in comparison with ATT. On the other hand, it spent more time in the Catch position because the activation was still high after the model reached the Catch position. Hence, there was no improvement in the number of strokes per minute (see bar charts from Figure 6.12). Figure 6.13 shows the quadriceps activation for ATT+AMP was smaller than the value for ATT during the second Drive state, but in general the average quadriceps activation increased (see bar charts from Figure 6.11). The nonzero average hamstrings activation also increased and was bigger than the nonzero average hamstrings activation of the controller with constant activation of 0.50. The maximum velocity was similar to the maximum velocity of ATT.

Davoodi and Andrews (2004) have used an FLC to reduce the muscle energy. In their study, hamstrings activation was maximum at the beginning of the Recovery state, and it decreased after. The maximum velocity achieved by both participants with SCI was between 0.59 m/s and 0.74 m/s.

ATT+AMP was able to decrease the activation during Drive and Recovery states. On the other hand, the hamstrings activation to exit the Handle Pull position increased, and the performance was not better than ATT. The use of a gain to scale the contribution of $(\dot{\theta}_m - \omega_{ref})$ is a possible solution to improve ATT+AMP.

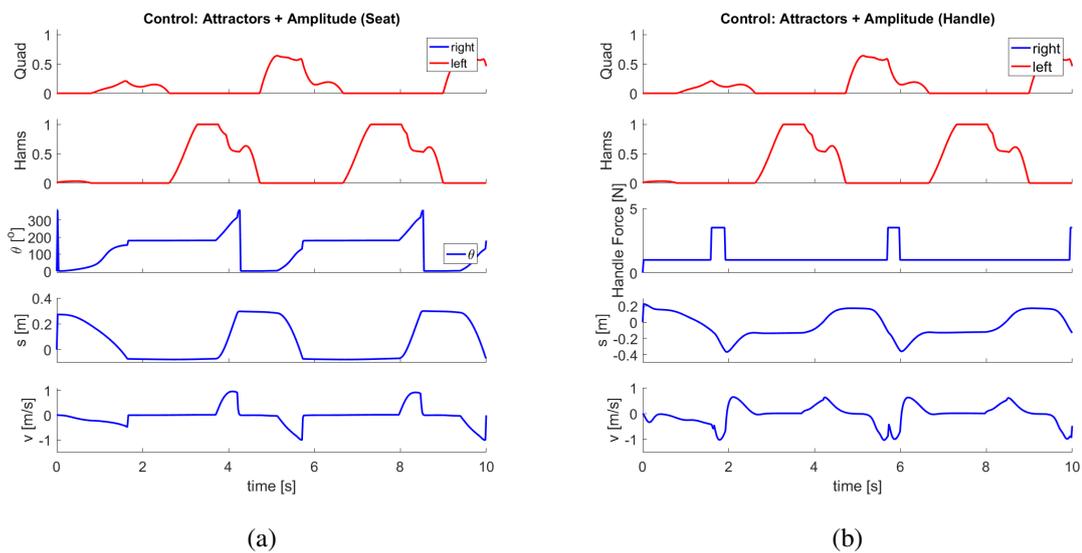


Figure 6.13: FES rowing simulation results for controller based on attractor dynamics with amplitude control. The graphs show the measures of (a) the seat and (b) the handle.

6.2.4 Addition of coupled oscillators

6.2.4.1 Method

All previous controllers have the same characteristic: the muscle groups are kept activated until the model reaches the desired position. In controllers based on attractor dynamics, sometimes the activation remained because x and y still had to achieve the value of the initial or target state. As a consequence, the model did not decrease the velocity until it suddenly stopped because of the safety stop simulated by the *CoordinateLimitForce*. The interruption of the movement can be seen in Figures 6.10 and 6.13.

The inertia caused by abrupt stopping can damage the joints of a participant. Controllers which minimize the position or velocity error reduce the muscle group activation, but this is not enough to lessen the velocity before the model achieves the desired position. A better option would be activating the antagonist muscle to speed up velocity decrease. We implemented this feature controlling the Hopf oscillator frequency with coupled oscillators. The new frequency is

$$\theta_{c1} = \text{atan2}(y, x), \quad (6.13)$$

$$\dot{\theta}_{c1} = \omega_{ref} + a_{c1} \sin(\theta_m - \theta_{c1}), \quad (6.14)$$

where ω_{ref} is the desired velocity, a_{c1} the coupling constant, θ_m the actual angle and θ_{c1} the angle of the controller. Note that when θ_m is greater than θ_{c1} , the frequency decreases, and it increases otherwise. An interesting feature of this controller is that it increases the frequency $\dot{\theta}_{c1}$ when the seat velocity v is too high, and the difference $\theta_m - \theta_{c1}$ becomes greater than 0° . Then, instead of maintaining a high amplitude until the model gets close to the desired position, the Hopf oscillator increases the oscillation until it changes the signal of x , i.e., the muscle group that is activated changes. Hence, the activation of the antagonist decreases the velocity before the model reaches the desired position. The previous feedback could not increase the frequency.

Additionally, it was possible to use the original equations of the inputs S_i and S_t due to s_n . The inputs to reach the Handle Pull position are

$$S_i = 0, \quad (6.15)$$

$$S_t = 1 - (\tanh(10(s_n - 0.7 \cdot (-1))) + 1)/2, \quad (6.16)$$

and the inputs to reach the Catch position are

$$S_i = 1 - (\tanh(10(0.7 \cdot (1) - s_n)) + 1)/2, \quad (6.17)$$

$$S_t = 0. \quad (6.18)$$

We used GA to find the parameters a , b , ω_{ref} and a_{c1} . The values are shown in Table 6.3.

6.2.4.2 Results and analysis

Simulation results for the modified controller based on attractors dynamics (ATT+OSC) are shown in Figure 6.14. It is possible to see a notable decrease in velocity during Recovery state. The quadriceps activation started before the model has reached the Catch position. However, the velocity during Drive state did not decrease because the Target neuron spent much time active, and the coupled oscillators only work during oscillation. Note that $\theta_{c1} = 0^\circ$ during handle pulling in Figure 6.14.b, i.e., $y = 0$. ATT+OSC achieved the Catch position after the Recovery state, but it started the Drive state soon after. Additionally, the model took a long time to exit the Handle Pull position during Recovery state. Note that the frequency of the oscillation is reduced during the beginning of the Recovery state, i.e., θ_{c1} is greater than θ_m .

We added the amplitude control (ATT+OSC+AMP) to increase the hamstring activation during the beginning of the Recovery state. Simulation results are shown in Figure 6.15. It is possible to see that the activation level increased but the activation of quadriceps before reach the Catch position was preserved. However, the reduction of the velocity was not as effective as the reduction of ATT+OSC.

The bar charts of Figure 6.11 show that ATT+OSC had the smallest nonzero average quadriceps activation and nonzero average hamstrings activation, which were 0.23 and 0.29, respectively. Different from ATT+AMP, ATT+OSC+AMP achieved a nonzero average hamstrings activation smaller than the controller with constant activation, and it was also smaller than the value for ATT. The maximum velocity of ATT+OSC was equal to the maximum velocity of the controller with constant activation of 0.50 (see the bar chart of Figure 6.12.a), while the nonzero average activations were almost half. ATT+OSC+AMP achieved a maximum velocity of 0.80 m/s which was high but not greater than the values of ATT and ATT+AMP. On the other hand, the bar chart of Figure 6.12.b shows that the number of strokes per minute for ATT+OSC and ATT+OSC+AMP was bigger than the number of strokes per minute of the controller with constant activation of 0.50. ATT+OSC+AMP also had a similar result when compared with the simulation in Hussain et al. (2011), even though the handle pulling did not occur in the Recovery state.

The addition of coupled oscillators solved the problem of the sudden stop, but only for the Catch position. The velocity during Drive state did not decrease because the Target neuron spent much time active, and the coupled oscillators only work during oscillation. Additionally, the addition of the amplitude control improved the performance, but at the cost of a high activation level. However, the cost is low compared to previous controllers, and the performance is similar to other studies (Davoodi et al., 2002; Hussain et al., 2011; Ektas et al., 2014).

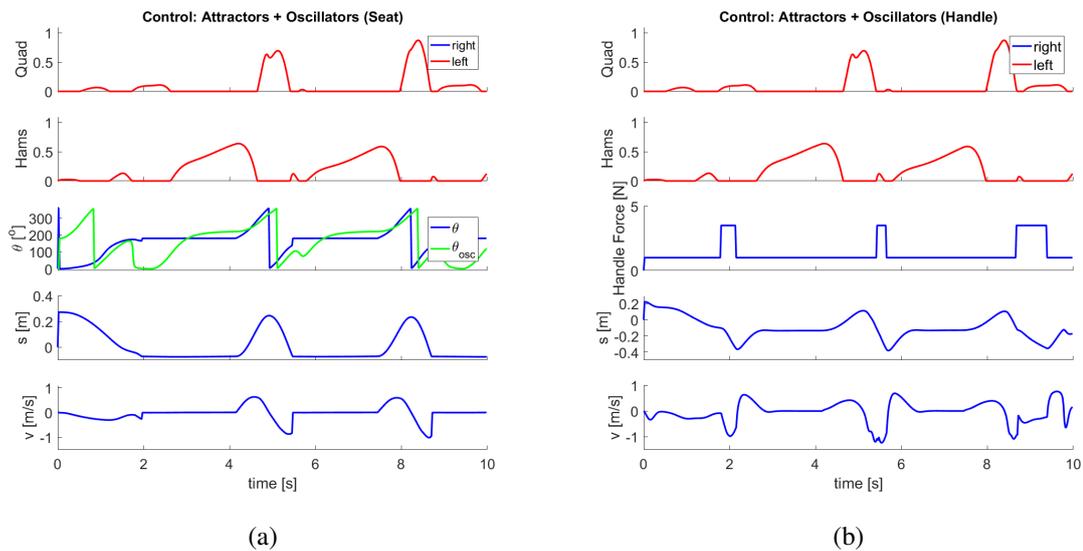


Figure 6.14: FES rowing simulation results for controller based on attractor dynamics with coupled oscillators. The graphs show the measures of (a) the seat and (b) the handle. θ_{OSC} corresponds to σ .

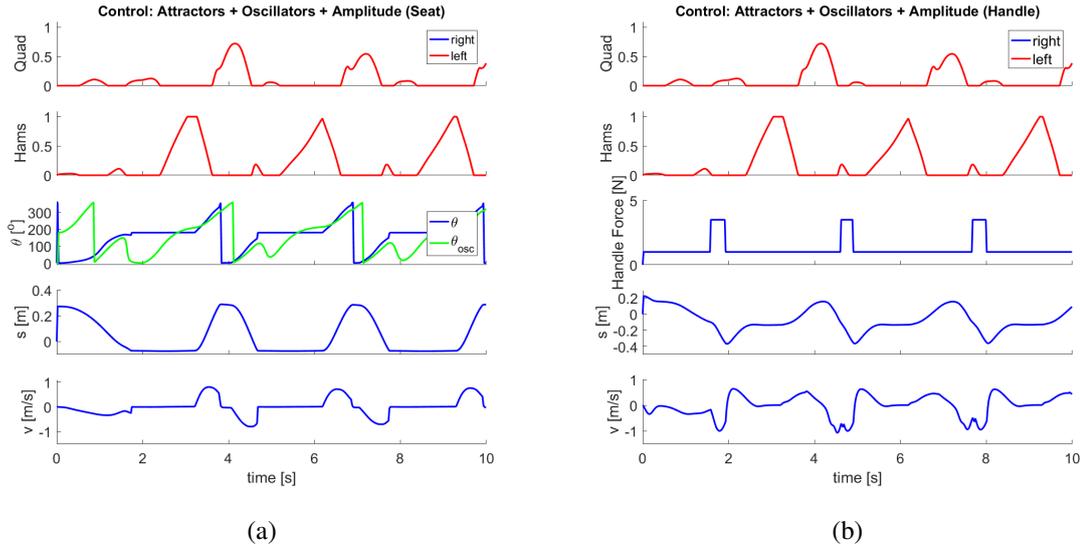


Figure 6.15: FES rowing simulation results for controller based on attractor dynamics with coupled oscillators and amplitude control. The graphs show the measures of (a) the seat and (b) the handle. θ_{OSC} corresponds to σ .

6.3 MUSCLE WORK RATE AND ENERGY COST EFFICIENCY

Considering one of the major benefits of FES-assisted physical exercise involves cardiovascular health, we decided to use muscle work rate and energy cost efficiency to compare FES applications since both cycling and rowing produce mechanical work and the movements have a metabolic energy cost. The measures for ATT, ATT+AMP, ATT+OSC, and ATT+OSC+AMP during the interval from 4 s to 9 s are shown in Figures 6.17. This interval has the same length of the interval used for FES cycling. As the musculoskeletal models used for FES cycling and FES rowing have different numbers of muscles, we used only the muscles used by the FSC. The energy expenditure of the muscles of the arms was not calculated, since the model does not have these muscles, and the results are compared with FES cycling.

The energy cost contains intervals with high peaks, and with low energy (the late occurs when the model did not move). When the model was in the Handle Pull position, which is evidenced by the large interval of work rate near 0 W/kg, the energy cost increased. The increase happened because of the heat generated when a high activation was applied to exit the Handle Pull position at the beginning of the Recovery state. It is more notable for controllers without coupled oscillators (see Figure 6.17), since their energy cost seems to reach a maximum limit before the recovery starts. Also, notice that the peaks of energy cost and work rate for controllers with coupled oscillators are smaller than the values for the other controllers. ATT achieved energy cost close to 250 W/kg, and work rate close to 200 W/kg. Despite the fact that the work rate during Drive state (i.e., before the large interval of work rate near 0 W/kg) reduced smoothly in simulations with controllers with coupled oscillators, in general, the slope of the curve during Drive and Recovery states was significantly accentuated. Hence, FES rowing requires a significant power from the muscles.

The average energy cost efficiency for FES rowing controllers is shown in Figure 6.16. It is important to know that the controllers achieved different velocities, and the motion is not a complete steady-state motion. The energy cost efficiencies of ATT and ATT+OSC was better than the other controllers. Since ATT+OSC had an average activation smaller than ATT and more strokes per minute than ATT, we can say that the addition of coupled oscillators improved the energy cost efficiency of FES rowing simulations. On the other hand, the

use of an amplitude control seems to reduce the energy cost efficiency, even though ATT+OSC+AMP had the highest number of strokes per minute.

If we include the general performance, a controller based on attractor dynamics and coupled oscillators proved to be a good alternative method to improve real tests. Also, the amplitude control needs to be improved to achieve a better efficiency without reducing the performance.

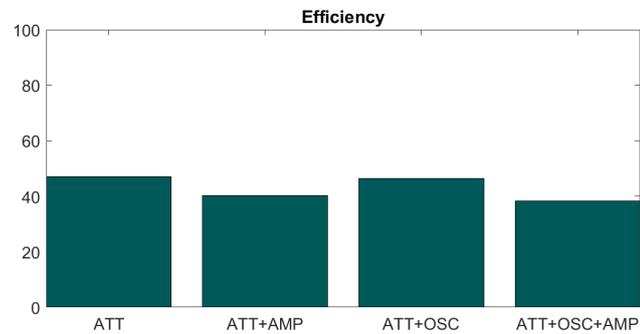


Figure 6.16: Energy cost efficiency of FES rowing with (a) controller based on attractor dynamics, (b) controller based on attractor dynamics with amplitude control, (c) controller based on attractor dynamics with coupled oscillators, and (d) controller based on attractor dynamics with coupled oscillators and amplitude control. The average energy cost efficiency was defined as the average total work over the average total metabolic energy cost (Section 4.5).

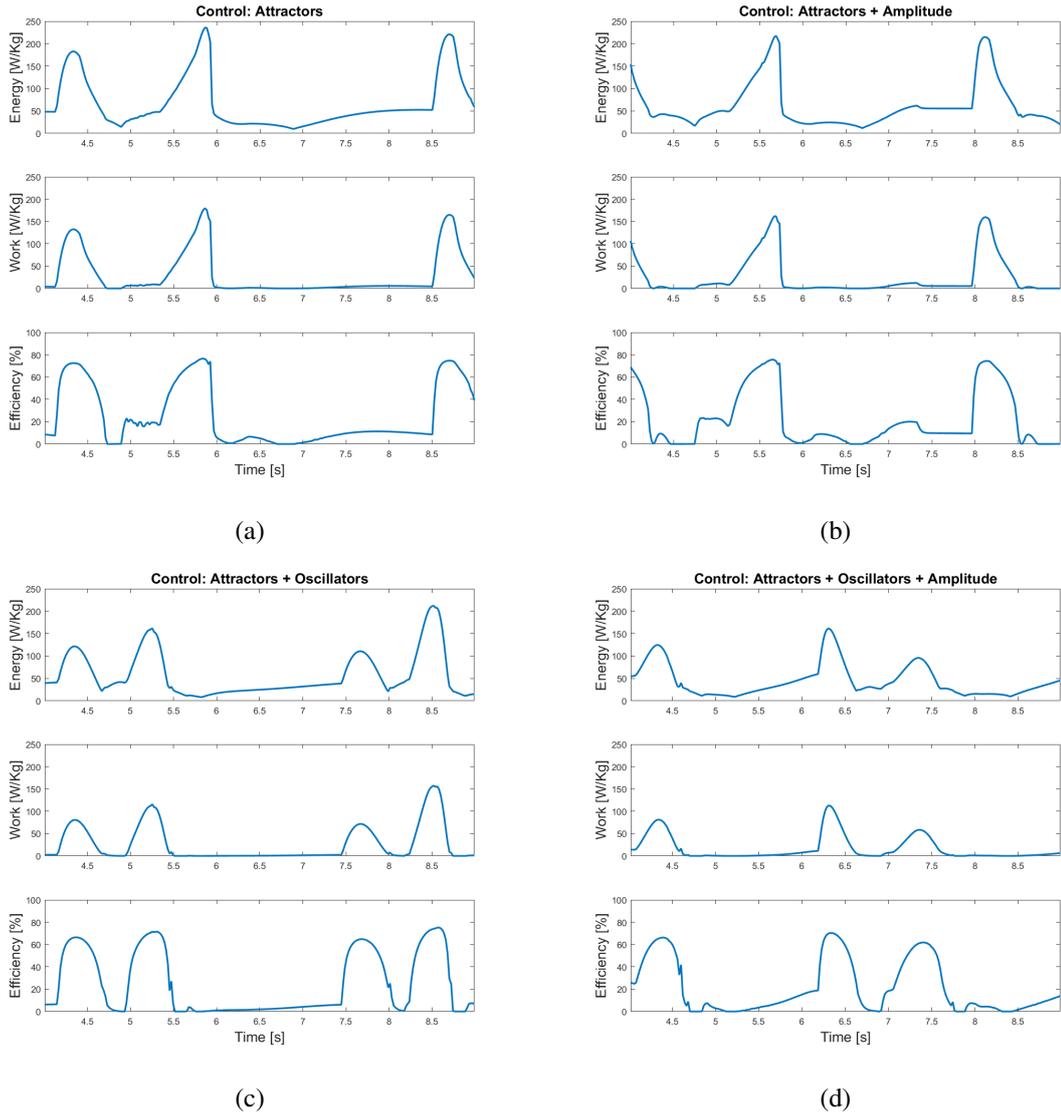


Figure 6.17: Metabolic energy cost, mechanical work rate and energy cost efficiency of FES rowing with (a) controller based on attractor dynamics, (b) controller based on attractor dynamics with amplitude control, (c) controller based on attractor dynamics with coupled oscillators, and (d) controller based on attractor dynamics with coupled oscillators and amplitude control. The energy is expressed in W/kg of muscle tissue.

7

DISCUSSION

This work was developed based on OpenSim. OpenSim provided default models for lower limbs simulations with a different number of degrees of freedom and muscle-tendon actuators and tools to analyze the results. Its compatibility with the *.obj* format and the package for Matlab made the addition of new objects to the models easy. All models we developed have loops in the kinematics, which was possible due to Constraints.

The cyclic motion was divided into states, and we used an FSC to choose which muscles will be activated by the lower level controller. In the case of FES cycling, we used angle intervals as states. A finite state machine was used for controllers with constant activation in FES rowing, while the controllers based on attractor dynamics already had a control layer. We did not explore the plasticity of neurons (Gerstner and Kistler, 2002) in this work, but a control layer of neurons can adapt its parameters to improve the performance. A control layer of neurons can also be used to modulate the angle intervals of the FSC for FES cycling.

It is important to notice that GA was applied only for simulations with step input and without environment modifications. Besides the time cost, the objective in using GA was to find a solution good enough to perform a stable motion even with environment modifications.

In the first section of this chapter, we discuss the limitations of using musculoskeletal models. The next sections present a general analysis of the results of FES cycling and FES rowing. The chapter concludes with the comparison of the FES exercises through the efficiency of the muscles energy expenditure.

7.1 MODEL LIMITATIONS

As the time cost increase with the complexity of the musculoskeletal model, we used a primarily lower extremity model with 10 degrees of freedom and 18 muscle-tendon actuators for FES cycling simulations. However, this model does not feature arms. Hence we had to use an expanded version of the lower extremity model for FES rowing simulations, which contains 29 degrees of freedom and 92 muscle-tendon actuators, including arms with inertial properties (Hamner et al., 2010). Due to the time cost, simulations for FES rowing had only 10 s, and environment modifications were not used.

An FES cycling simulation takes approximately 15 minutes to finish. An FES rowing simulation takes four times this period. To run a GA with a population size of 10 and a maximum number of generations of 10, it takes approximately one day and four days, respectively. We used a machine with Intel(R) Core(TM) i5-3230M CPU (2.60GHz), 8GB RAM, and based on x64 architecture.

The new objects were added to the musculoskeletal models as part of the kinematics. This method was possible due to the addition of loops using Constraints, which are similar to joints. Then, there are no reaction forces between feet and machines. A force reaction measure could allow the creation of control strategies based on force effectiveness (Bini et al., 2013). An alternative is to attach the feet again to the pedals using physical contact, even though it increases the cost of the simulations.

Additionally, the improvised chain made with a muscle model for FES rowing had its limitations. It was possible to use the handle position to start and finish the pulling, but the handle position and velocity could not be used as feedback. The elastic properties of the muscle helped to simulate the handle pulling. It was possible to cancel the constant force of the force actuator before the joints exceed their limits. However, the elastic properties also affected the velocity of the handle (see Figure 6.15.b). A better method for the next version of the model could be a force actuator with a damping factor together with a *CoordinateLimitForce* to avoid

breaking the kinematics.

At last, some simulations for FES cycling did not complete. These simulations had in common that the model presented high accelerations. Longer simulations were also unstable and had the same problem when the controller was active for more than 50 s. Simulations for FES rowing also presented instability when the arms moved fast. Since no forces were acting in the arms directly, they end up exceeding the joint limits. Hence, the models need improvements for a stable motion during simulations, perhaps changing numerical integration properties.

7.2 FES CYCLING

The strategy of activating the opposing muscle to reduce the velocity and tune the low-level controller using GA to be active most of the time during simulation proved to be effective. The maximum error and RMSE in steady state reduced when compared with other control strategies, and the nonzero average activation was also smaller. The rise time was around 40-60 ms, which does not represent a transient response of a muscle assisted by FES. In general, OpenSim does not incorporate properties that allow analyzing the transient response of FES systems.

The strategy of activating the opposing muscle also had better performance than studies found in state of the art. However, we used OpenSim and a different tricycle design in simulations with different configurations. Hence, it is hard to compare performances and decide which is best.

Simulations demonstrated that FLC is more stable with models with different scales, i.e., the performance does not tend to change much with different heights. Hence, FLC is more suitable for real tests with a large population. On the other hand, the PI controller presented a better performance with the model used to tune the parameter. It can be used, for example, for a high performance training for athletes with paraplegia. The PI controller was also more robust in simulations with small fatigue, but the performance of other controllers with the gluteus muscle group was similar.

The addition of coupled oscillators for the PI control did not seem to improve the performance. We did not find studies about the minimization of both angular velocity error and angle error. Additionally, results from simulations are not enough to evaluate if there is any health benefit or if it increases the time of real tests. However, it is important to note that the controller with coupled oscillators could couple with the ideal controller or follow it with a constant difference phase in simulations with different loads, except when the torque was too big. As a consequence, the maximum errors and RMSEs were smaller than the values obtained with PI control alone. Figures 5.22.a and 5.22.b show that the mean angular velocity for PI controller alone is below the desired velocity while the mean angular velocity for PI controller with coupled oscillators is closer.

Even though the musculoskeletal model is not as detailed as a person, the parameters we found for the controllers can be used as a start point to find better values for the participants. Additionally, as the real tricycle has load on the wheels and we expect to use the system with different participants, we can add coupled oscillators in the FLC to achieve better results in real test with a large population, since this combination can improve the performance of FLC in simulations with different loads while preserving its performance with different scales.

7.3 FES ROWING

Instead of the four finite states used in other works, the controller based on attractor dynamics divided the rowing motion into two discrete movements and one oscillation. Hence, it was possible to control the oscillatory part as a rhythmic movement. Also, the muscle activation during discrete movements was 0.1, while some studies have used maximum muscle force (Davoodi et al., 2002; Yusoff et al., 2014). Even if 0.1 is not enough, the attractors will increase the activation when the seat moves out of position.

Even though the model achieved a maximum velocity greater than the controller with constant activation of 0.50, the addition of feedback to control frequency and amplitude of the oscillation were not enough to reduce the average activation level or increase the number of strokes per minute. Furthermore, the model abruptly stops when it achieves a safety stop. We used coupled oscillators to activate the antagonist muscles before the model reaches the safety stop. As a result, it is possible to see a notable decrease in velocity during Recovery state (see Figure 6.15). However, the velocity during Drive state did not decrease because the Target neuron spent much time active, and the coupled oscillators only work during oscillation.

The controller with coupled oscillators and amplitude control achieved a better performance than the controller with constant activation of 0.50, while the average muscle activation was small than 0.50. The number of strokes per minute was also similar to the results of simulations found in state of the art. We are planning to test the developed controller in simulations with different environment modifications and different scales. Additionally, the implementation in a real ergometer machine is necessary to see if the control strategy can achieve the performance of strategies which produce maximum muscle force. The parameters we found for the controllers can be used as a start point to find better values for the participants. We expect that the low muscle activation will lessen the fatigue effect and increase the time of real tests.

7.4 ENERGY COST EFFICIENCY OF RHYTHMIC FES APPLICATIONS

Despite the fact that we could not calculate an efficiency based on Hunt et al. (2012), the OpenSim tool based on Umberger et al. (2003) made possible the comparison between FES cycling and FES rowing. The high values found, especially for FES rowing, are due to the muscle energy expenditure expressed in W/kg of muscle tissue. The energy expenditure of a person doing an exercise is usually expressed in W/Kg of the whole body weight. Hence, if we consider for example that the eight muscles used to perform the rowing motion weigh 3 kg, a total energy expenditure of 230 W/kg in the simulations may be equivalent to 9.18 W/kg for a person who weighs 75.16 kg. Nevertheless, it is still possible to compare results between simulations, since we used the same muscle energy expenditure model for all of them.

The addition of coupled oscillators improved the energy cost efficiency of FES exercises. If we include the general performance, it proved to be a useful alternative method to improve real tests. In the case of FES rowing, the use of an amplitude control seems to reduce the energy cost efficiency, even with coupled oscillators. The characteristics of the cycling performance include a small muscle activation and a cadence of 33 rpm in the interval between 14 s and 19 s. The rowing exercise had a higher muscle activation and the number of strokes per minute was less than 20 strokes/min.

The bar charts of Figures 5.43 and 6.16 show that FES rowing has a better energy cost efficiency than FES cycling, even though the model stopped during the discrete movements. However, analyzing the energy cost and the work rate separately, FES rowing had high peaks of energy and the slope of the curve during Drive and Recovery states was significantly accentuated. On the other hand, FES cycling produced less heat and

had smooth variations of energy. Hence, FES cycling is more suitable for an initial physical conditioning. The participants start a protocol with weak muscles and low endurance, so an exercise which allows a longer training is preferable. In experimental studies, FES cycling sessions lasted a few minutes to an hour (Hunt et al., 2002). FES rowing sessions lasted from 2 to 5 minutes (Davoodi et al., 2002; Draghici and Shefelbine, 2014), but they were focused on getting performance measures of the controllers and did not analyze the performance of the participant.

After the initial physical conditioning, FES rowing can be used to improve the power of the muscles. Additionally, FES rowing needs constant concentration and synchronization of the voluntary upper body motion with the FES assisted leg motion. This cooperation of the CNS increases the chance of enhancing the excitability and facilitate plasticity of the neural structures in the damaged area of the spinal cord. Another possibility is finding an alternative neural pathway the send commands to the muscles. Hence, FES rowing can be used as a late stage for physical conditioning and an early stage of the rehabilitation process.

The objective of this project was to explore the characteristics of rhythmic control strategies to improve the performance of FES cycling and FES rowing as part of a broader research effort targeting SCI rehabilitation. We developed a simulation environment with detailed musculoskeletal models to accelerate the prototyping of new control strategies.

It was possible to simulate muscle activation through closed-loop control. Despite the simplicity of the models when compared with the human body, the parameters we found for the controllers can be used as a start point to find better values for the participants. However, the musculoskeletal models still need improvements, since some simulations stopped working correctly. The instability seems to be related to high accelerations. Hence, some precautions are necessary to avoid significant accelerations during simulations.

The basic control for both applications was divided into two layers: a higher level finite state controller which decides the muscles that will be activated, and a lower level controller which calculates the activation level. In the case of FES cycling, we used the same parameters found using genetic algorithm in simulations with noise, fatigue, different loads, and scales in order to evaluate the stability of the controllers.

The addition of coupled oscillators to the PI controller for FES cycling granted the minimization of angle error, and the performance improved in the cases the PI controller alone could not reach the desired velocity (e.g., simulations with different loads). A new study is necessary to evaluate if the addition of coupled oscillators in the fuzzy logic controller would achieve better results for real tests with a large population.

The controller based on attractor dynamics proved to be an alternative control for FES rowing. Instead of a finite state controller, it uses a neural network to decide which motor primitive will be activated. It was possible to increase the number of strokes per minute adding coupled oscillator. Additionally, the coupled oscillators reduced the abrupt change of the velocity in the safety stops.

FES cycling controllers used less muscle activation, and the model spent less metabolic energy to perform the motion than FES rowing controllers. Hence, FES cycling is more suitable for participants with weak muscles and low endurance. Furthermore, the addition of coupled oscillators improved the energy cost efficiency of the exercises. Simulation results indicate that FES cycling can be used first for physical conditioning. FES rowing can be utilized later, especially because of the synchronization of the voluntary upper body motion and the stimulated legs motion. For that reason, FES rowing can also help with rehabilitation besides the physical conditioning.

In the future, we plan to polish the musculoskeletal models and basic framework. A new study is necessary to evaluate if the addition of coupled oscillators in the fuzzy logic controller would achieve better results for real tests with a large population.

Instead of a complex function, we are planning to use a constant load to evaluate the performance of the controllers for FES cycling. Additionally, we will research in state of the art for a more sophisticated fatigue model. For FES rowing, we plan to compare the controller based on attractor dynamics with other controllers which reduce muscle activation level and test a variety of environment configurations. Furthermore, we will add muscles to the arms and use the handle position as a feedback for the lower level controllers.

We can go further and explore the plasticity of neurons to make a controller which adapts its parameters according to the musculoskeletal model. Also, instead of using only a control layer composed of neurons, it is possible to use neurons for the whole controller. For example, we can change the nonuniform oscillators for neural oscillators.

The basic framework can also be used for other rhythmic activities. Most equipment used in a gym is

based on cyclic movement to allow series of repetitions, which resembles the movements investigated in this work. Additionally, each equipment is specific for a group of muscles, which may not be targeted with other equipment. Following the same process of the development of the musculoskeletal models for cycling and rowing, we can use Blender to create new bodies, add them to the default musculoskeletal model using Matlab, and lock some degrees of freedom to simulate a new exercise. The basic framework will allow the measure of performance, the addition of environment modifications, and comparison of the efficiency. Then, we can evaluate which exercises are more suitable for physical conditioning of specific muscles, weak muscles, low endurance, frequent spasms, and other characteristics specific of people with SCI.

Besides the possibilities of future work mentioned above, our main objective in the near future is testing the new controllers in the real systems. The parameters of the controllers found by the genetic algorithm will be a start point to the parameters for the participants, and we will also test different levels of the load at least with two participants with different height and weight. It will also be possible to observe the performance during longer tests.

Bibliography

- Abdulla, S. C., O. Sayidmarie, and M. O. Tokhi (2014). Functional electrical stimulationbased cycling assisted by flywheel and electrical clutch mechanism: A feasibility simulation study. *Robotics and Autonomous Systems* 62(2), 188–199.
- Abdulla, S. C. and M. O. Tokhi (2014). Optimization of indoor fescycling exercise assisted by a flywheel mechanism using genetic algorithm. *2014 IEEE International Symposium on Intelligent Control (ISIC)*, 1867–1871.
- Angeli, C. A., V. R. Edgerton, Y. P. Gerasimenko, and S. J. Harkema (2014). Altering spinal cord excitability enables voluntary movements after chronic complete paralysis in humans. *Brain* 137(5), 1394–1409.
- Arnold, P. B., P. P. McVey, W. J. F. T. M. Deurloo, and A. R. Grasso (1992). Functional electric stimulation: its efficacy and safety in improving pulmonary function and musculoskeletal fitness. *Archives of physical medicine and rehabilitation* 73(7), 665–668.
- Bellman, M. J., T.-H. Cheng, and R. J. D. W. E. Dixon (2014). Cadence control of stationary cycling induced by switched functional electrical stimulation control. *2014 IEEE 53rd Annual Conference on Decision and Control (CDC)*, 6260–6265.
- Berkelmans, R. (2008). Fes cycling. *Journal of Automatic Control* 18(2), 73–76.
- Bini, R., P. Hume, J. L. Croft, and A. Kilding (2013). Pedal force effectiveness in cycling: a review of constraints and training effects. *Cycling Research Center*.
- Blender, Online Community (2013). *Blender - a 3D modelling and rendering package*. Blender Institute, Amsterdam: Blender Foundation.
- Bó, A. P. L., L. O. da Fonseca, J. A. Guimarães, E. FachinMartins, M. E. G. Paredes, G. A. Brindeiro, A. C. C. de Sousa, M. C. N. Dorado, and F. M. Ramos (2017, Dec). Development of an fescycling system and preparation of an individual with sci for the cybathlon race 2016. *IEEE Robotics and Automation Magazine*.
- Chen, J.-J. J., C.-T. Shih, D.-G. Huang, N.-Y. Yu, M.-S. Ju, and T.-C. Huseh (1994). Development of fescycling system with closedloop control. *Chinese Journal of Medical and Biological Engineering* 14(3), 195–208.
- Chen, J.-J. J., N.-Y. Yu, D.-G. Huang, B.-T. Ann, and G.-C. Chang (1997). Applying fuzzy logic to control cycling movement induced by functional electrical stimulation. *IEEE transactions on rehabilitation engineering* 5(2), 158–169.
- Chen, K., S.-C. Chen, K.-H. Tsai, J.-J. J. Chen, N.-Y. Yu, and M.-H. Huang (2004). An improved design of home cycling system via functional electrical stimulation for paraplegics. *International journal of industrial ergonomics* 34(3), 223–235.
- Coley, D. A. (1999). *An introduction to genetic algorithms for scientists and engineers*. World Scientific Publishing Co Inc.
- da Fonseca, L. O., A. P. L. Bó, J. A. Guimarães, M. E. G. Paredes, U. de Araújo, C. AzevedoCoste, and E. FachinMartins (2016). Cadence tracking and disturbance rejection in fes cycling for paraplegic subjects: a case study. *European Journal of Translational Myology*.
- Davoodi, R. and B. J. Andrews (2003). Switching curve control of functional electrical stimulation assisted rowing exercise in paraplegia. *Medical and Biological Engineering and Computing* 41(2), 183–189.

- Davoodi, R. and B. J. Andrews (2004). Fuzzy logic control of fes rowing exercise in paraplegia. *IEEE Transactions on Biomedical Engineering* 51(3), 541–543.
- Davoodi, R., B. J. Andrews, and G. D. Wheeler (2001). Modified concept 2tm rowing machine with manual fes controller for total body exercise in paraplegia. *Proceedings of International FES Society*, 114–116.
- Davoodi, R., B. J. Andrews, and G. D. Wheeler (2002). Automatic finite state control of fes-assisted indoor rowing exercise after spinal cord injury. *Neuromodulation: Technology at the Neural Interface* 5(4), 248–255.
- Davoodi, R., B. J. Andrews, G. D. Wheeler, and R. Lederer (2002). Development of an indoor rowing machine with manual fes controller for total body exercise in paraplegia. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 10(3), 197–203.
- de Sousa, A. C. C., F. M. Ramos, M. C. N. Dorado, L. O. da Fonseca, and A. P. L. Bó (2016). A comparative study on control strategies for fes cycling using a detailed musculoskeletal model. *IFAC-PapersOnLine* 49(32), 204–209.
- Degallier, S. and A. Ijspeert (2010). Modeling discrete and rhythmic movements through motor primitives: a review. *Biological cybernetics* 103(4), 319–338.
- Deley, G., J. Deneziller, and N. Babault (2014). Functional electrical stimulation: cardiorespiratory adaptations and applications for training in paraplegia. *Sports Medicine* 45(1), 71–82.
- Delp, S. L., F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, C. T. John, E. Guendelman, and D. G. Thelen (2007, Nov). Opensim: Open-source software to create and analyze dynamic simulations of movement. *IEEE Transactions on Biomedical Engineering* 54(11), 1940–1950.
- Delp, S. L., J. P. Loan, M. G. Hoy, F. E. Zajac, E. L. Topp, and J. M. Rosen (1990). An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical engineering* 37(8), 757–767.
- Donati, A. R. C., S. Shokur, E. Morya, D. S. F. Campos, R. C. Moiola, C. M. Gitti, P. B. Augusto, S. Tripodi, C. G. Pires, G. A. Pereira, F. L. Brasil, S. Gallo, A. A. Lin, A. K. Takigami, M. A. Aratanha, S. Joshi, H. Bleuler, G. Cheng, A. Rudolph, and M. A. L. Nicoletis (2016). Long-term training with a brain-machine interface-based gait protocol induces partial neurological recovery in paraplegic patients. *Scientific reports* 6, 30383.
- Draghici, A. E. and S. J. Shefelbine (2014). Assessing kinetics of fes rowing in spinal cord injury patients. *40th Annual Northeast Bioengineering Conference (NEBEC)*, 1–2.
- Eichhorn, K. F., W. Schubert, and E. David (1984). Maintenance, training and functional use of denervated muscles. *Journal of biomedical engineering* 6(3), 205–211.
- Ektas, N., N. M. Yusoff, R. Smith, M. Halaki, N. A. Hamzaid, N. Hasnan, and G. M. Davis (2014). Development of a sliding stretcher fes-rowing system. *IEEE 19th International Functional Electrical Stimulation Society Annual Conference (IFESS)*, 1–3.
- Fornusek, C., G. M. Davis, and M. F. Russold (2013, May). Pilot study of the effect of low-cadence functional electrical stimulation cycling after spinal cord injury on thigh girth and strength. *Archives of Physical Medicine and Rehabilitation* 94(5), 990–993.
- Fornusek, C., P. J. Sinclair, and G. M. Davis (2007, January). The forcevelocity relationship of paralyzed quadriceps muscles during functional electrical stimulation cycling. *Neuromodulation : journal of the International Neuromodulation Society* 10(1), 68–75.

- Gerasimenko, Y. P., D. C. Lu, M. Modaber, S. Zdunowski, P. Gad, D. G. Sayenko, E. Morikawa, P. Haakana, A. R. Ferguson, R. R. Roy, and V. R. Edgerton (2015). Noninvasive reactivation of motor descending control after paralysis. *Journal of neurotrauma* 32(24), 1968–1980.
- Gerstner, W. and W. M. Kistler (2002). *Spiking Neuron Models: Single Neurons, Populations, Plasticity*. Cambridge University Press.
- Gföhler, M. and P. Lugner (2004, December). Dynamic simulation of fes-cycling: Influence of individual parameters. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 12(4), 398–405.
- Glaser, R. M., J. A. Gruner, S. D. Feinberg, and S. R. Collins (1983). Locomotion via paralyzed leg muscles: feasibility study for a leg-propelled vehicle. *Journal of rehabilitation R&D* 20(1), 87–92.
- Hamner, S. R., A. Seth, and S. L. Delp (2010, Oct). Muscle contributions to propulsion and support during running. *Journal of Biomechanics* 43(14), 2709–2716.
- Harkema, S., Y. Gerasimenko, J. Hodes, J. Burdick, C. Angeli, Y. Chen, C. Ferreira, A. Willhite, E. Rejc, R. G. Grossman, and V. R. Edgerton (2011). Effect of epidural stimulation of the lumbosacral spinal cord on voluntary movement, standing, and assisted stepping after motor complete paraplegia: a case study. *The Lancet* 377(9781), 1938–1947.
- Hase, K., B. J. Andrews, A. B. Zavatsky, and S. E. Halliday (2002). Biomechanics of rowing. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing* 45(4), 1082–1092.
- Hunt, K. J., J. Fang, J. Saengsuwan, M. Grob, and M. Laubacher (2012). On the efficiency of fes cycling: A framework and systematic review. *Technology and Health Care* 20(5), 395–422.
- Hunt, K. J., J. Fang, T. Schauer, N.-O. Negård, W. Stewart, and M. H. Fraser (2002). A pilot study of lowerlimb fes cycling in paraplegia. *Proc. 7th Annu. Conf. Int. Functional Electrical Stimulation Soc.*
- Hunt, K. J., M. Rothe, T. Schauer, A. Ronchi, and N.-O. Negård (2001, January). Automatic speed control in fes cycling.
- Hussain, Z., M. O. Tokhi, and S. Gharooni (2008). Dynamic simulation of indoor rowing exercise for paraplegics. *Second Asia International Conference on Modeling & Simulation, 2008. AICMS 08*, 901–904.
- Hussain, Z., M. O. Tokhi, R. Jailani, and F. Ahmad (2010). Parameter optimization of fes-assisted indoor rowing exercise using moga. pp. 76–80.
- Hussain, Z., S. Z. Yahaya, R. Boudville, K. A. Ahmad, and M. H. M. Noor (2011). Self adaptive neuro-fuzzy control of fes-assisted paraplegics indoor rowing exercise. pp. 7–11.
- Kim, C.-S., G.-M. Eom, K. Hase, G. Khang, G.-R. Tack, J.-H. Yi, and J.-H. Jun (2008). Stimulation pattern-free control of fes cycling: Simulation study. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38(1), 125–134.
- Kralj, A. R. and T. Bajd (1989). *Functional electrical stimulation: standing and walking after spinal cord injury*. CRC press.
- Laskin, J. J., E. A. A. , L. M. O. , R. Burnham, D. C. Cumming, R. D. Steadward, and G. D. Wheeler (1993). Electrical stimulation-assisted rowing exercise in spinal cord injured people. a pilot study. *International Medical Society of Paraplegia* 31(8), 534–541.
- Lee, C.-C. (1990). Fuzzy logic in control systems: fuzzy logic controller. i. *IEEE Transactions on systems, man, and cybernetics* 20(2), 404–418.

- Li, P.-F., Z.-G. Hou, F. Zhang, Y.-X. Chen, X.-L. Xie, M. Tan, and H.-B. Wang (2010). Adaptive neural network control of fcs cycling. *2010 4th International Conference on Bioinformatics and Biomedical Engineering (iCBBE)*, 1–4.
- Li, P.-F., Z.-G. Hou, F. Zhang, M. Tan, H.-B. Wang, Y. Hong, and J.-W. Zhang (2009). An fcs cycling control system based on cpg. *31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS)*, 1569–1572.
- Loeb, G. E., F. J. R. Richmond, and L. L. Baker (2006). The bion devices: injectable interfaces with peripheral nerves and muscles. *Neurosurgical focus* 20(5), 1–9.
- Marieb, E. N. and K. Hoehn (2007). *Human anatomy & physiology*. Pearson Education.
- Matsuoka, K. (1985). Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Biological cybernetics* 52(6), 367–376.
- Mitchell, M. (1999). *An introduction to genetic algorithms*, Volume 3. MIT press, Fifth printing.
- Mutton, D. L., A. M. E. Scremin, T. J. Barstow, M. D. Scott, C. F. Kunkel, and T. G. Cagle (1997). Physiologic responses during functional electrical stimulation leg cycling and hybrid exercise in spinal cord injured subjects. *Archives of physical medicine and rehabilitation* 78(7), 712–718.
- Perko, L. (2013). *Differential equations and dynamical systems*, Volume 7. Springer Science & Business Media.
- Petrofsky, J. S., H. Heaton, and C. A. Phillips (1983). Outdoor bicycle for exercise in paraplegics and quadriplegics. *Journal of biomedical engineering* 5(4), 292–296.
- Popović, D. and T. Sinkjær (2012). *Control of movement for the physically disabled: control for rehabilitation technology*. Springer Science & Business Media.
- Schöner, G. and C. Santos (2001, Jul). Control of movement time and sequential action through attractor dynamics : a simulation study demonstrating object interception and coordination. *Intelligent Symposium on Intelligent Robotic Systems - SIRS'2001* 9.
- Stein, R. B. (1999). Functional electrical stimulation after spinal cord injury. *Journal of neurotrauma* 16(8), 713–717.
- Strogatz, S. H. (1994). *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. AddisonWesley Publishing.
- Umberger, B. R., K. G. M. Gerritsen, and P. E. Martin (2003). A model of human muscle energy expenditure. *Computer methods in biomechanics and biomedical engineering* 6(2), 99–111.
- van Mourik, A. M. and P. J. Beek (2004). Discrete and cyclical movements: unified dynamics or separate control? *Acta psychologica* 117(2), 121–138.
- Washburn, R. A. and S. F. Figoni (1999). High density lipoprotein cholesterol in individuals with spinal cord injury: the potential role of physical activity. *Spinal Cord* 37(10), 685–695.
- Yahaya, S. Z., Z. Hussain, R. Boudville, F. Ahmad, and M. N. Taib (2014). Optimization of flc parameters for optimal control of fcsassisted elliptical stepping exercise using ga and pso. *2014 IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 28–30.
- Yamaguchi, G. T. and F. E. Zajac (1989). A planar model of the knee joint to characterize the knee extensor mechanism. *Journal of biomechanics* 21(1), 1–10.

Yusoff, N. M., N. A. Hamzaid, N. Hasnan, N. Ektas, and G. M. Davis (2014). Automatic stimulation activation for a sliding stretcher fes-rowing system. *IEEE 19th International Functional Electrical Stimulation Society Annual Conference (IFESS)*, 1–3.