



INSTITUTO TECNOLÓGICO DE AERONÁUTICA

CONSELHO NACIONAL DE DESENVOLVIMENTO CIENTÍFICO E  
TECNOLÓGICO - CNPq



**PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA -  
PIBIC**

**INTEGRAÇÃO NEURAL UTILIZANDO DERIVADAS MÉDIAS EM ESTRUTURAS DE  
CONTROLE PREDITIVO PARA O CONTROLE DO PÊNDULO INVERTIDO NÃO LINEAR**

**Rahyan Azin Gondim Paiva**

**RELATÓRIO PARCIAL DE ATIVIDADES**

**Orientador: Paulo Tasinaffo**

**JANEIRO / 2015**



INSTITUTO TECNOLÓGICO DE AERONÁUTICA

CONSELHO NACIONAL DE DESENVOLVIMENTO CIENTÍFICO E  
TECNOLÓGICO - CNPq



**PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA -  
PIBIC**

**Relatório Parcial**

**INTEGRAÇÃO NEURAL UTILIZANDO DERIVADAS MÉDIAS EM ESTRUTURAS DE  
CONTROLE PREDITIVO PARA O CONTROLE DO PÊNDULO INVERTIDO NÃO LINEAR**

São José dos Campos, 25/ 01 / 2015

Nome do aluno	Rahyan Azin Gondim Paiva
Assinatura do aluno	<i>Rahyan Azin Gondim Paiva</i>

Nome do orientador	
Assinatura do orientador	

**INSTITUTO TECNOLÓGICO DE AERONÁUTICA**  
**PROGRAMA INSTITUCIONAL DE BOLSAS DE INICIAÇÃO CIENTÍFICA -**  
**PIBIC**

Formulário de Aprovação de Relatório pelo Orientador

**Relatório:**

☐

Rel. Parcial

☐

Rel. Final

---

**1- CONSIDERO O RELATÓRIO APROVADO COM BASE NOS SEGUINTE ASPECTOS**

---

---

**2- APRECIÇÕES DO ORIENTADOR SOBRE O DESEMPENHO DO BOLSISTA NA EXECUÇÃO DO TRABALHO DE INICIAÇÃO CIENTÍFICA**

---

---

**Local e data:**

**Assinatura do Orientador:**

## 1. Índice

Resumo do Plano Inicial .....	05
Resumo das Atividades Realizadas .....	06
Descrição do Problema .....	07
Resultados Obtidos .....	10
Conclusões .....	11
Plano de Trabalho e Cronograma das Etapas Seguintes .....	11
Bibliografia .....	12
Anexo A .....	13

## 2. Resumo de Plano Inicial

A inteligência artificial permitiu grandes avanços na área da computação. Uma das abordagens com maior sucesso são as redes neurais artificiais, que permitem, por exemplo, o reconhecimento da escrita, processamento de imagens, modelagem de sistemas dinâmicos não lineares, aplicações em teoria de controle e etc. Nesse trabalho pretende-se utilizar as redes neurais com arquitetura MLP (Multilayer Perceptron) numa estrutura de controle preditivo para resolver o problema de equilíbrio do pêndulo invertido não linear. A modelagem da dinâmica não linear será realizada utilizando-se o método das derivadas médias.

Uma boa introdução à teoria das redes neurais artificiais pode ser encontrada em Zurada, 1992, em Haykin, 1999 ou em Russell e Norvig, 2003. Um ponto de partida bastante importante no estudo das redes neurais artificiais é que elas são consideradas aproximadores universais de funções (e.g., Hornik et al, 1989; Cotter, 1990; Jang et al, 1997; Spooner et al, 2002). Assim, as redes neurais artificiais (RNA) têm sido muito utilizadas na modelagem de sistemas dinâmicos não lineares nas últimas décadas, pois elas possuem uma alta capacidade de aproximar mapeamentos não lineares. Vários estudos foram desenvolvidos nessa área utilizando a metodologia NARMAX (*Non Linear AutoRegressive Moving Average with exogenous inputs*) e posterior aplicação em controle (e.g., Narendra and Parthasarathy, 1990; Hunt et al., 1992; Narendra, 1996). Além dessa, outras metodologias também foram desenvolvidas nos últimos anos para representação de sistemas dinâmicos não lineares, entre elas: a metodologia das derivadas instantâneas (e.g., Wang e Lin, 1998, Rios Neto, 2001; Melo, 2008; Melo e Tasinaffo, 2010) e a metodologia das derivadas médias (Tasinaffo, 2003; Tasinaffo e Rios Neto, 2005 e 2006). Embora existam essas três metodologias para representação de sistemas dinâmicos não lineares, através de redes neurais artificiais com arquitetura *feedforward*, esse trabalho, como já mencionado no parágrafo inicial, se concentrará somente na metodologia das derivadas médias. A aplicação original do método das derivadas médias em uma estrutura de controle preditivo pode ser encontrada em Tasinaffo e Rios Neto, 2006.

## 3. Resumo das Atividades Realizadas

Nesse projeto, propõe-se a aplicação de redes neurais na resolução de equações diferenciais ordinárias não lineares e sua posterior aplicação em teoria de controle preditivo. Para o primeiro semestre foi proposto o estudo teórico necessário para a execução deste projeto, bem como a apresentação do cronograma da execução do projeto em si. Já no segundo semestre, com a base teórica necessária, adquirida no semestre anterior, será previsto o desenvolvimento e solução computacional numérica do problema de equilíbrio do pêndulo invertido não linear.

## 4. Descrição do Problema

O objetivo desse trabalho será resolver o problema de equilíbrio do pêndulo invertido não linear utilizando as redes neurais com arquitetura MLP (Multiplayer Perceptron). Será utilizado o método das derivadas médias para realizar a modelagem dinâmica não linear. Durante o primeiro semestre realizado o estudo teórico necessário para a execução do projeto.

### 4.1. Representação de Sistemas Dinâmicos de Redes Neurais Artificiais

Um sistema é dinâmico quando possui memória, ou seja, quando seu estado atual depende de condições passadas. As influências externas ao sistema e que não são afetadas por ele, são chamadas de *entradas* do sistema. O resultado da ação das influências externas e internas são descritos por indicadores chamados de *saídas* do sistema.

Para compreender os relacionamentos do tipo causa-efeito, é necessário então definir o sistema, suas entradas e saídas e realizar um processo de análise formal. É estabelecido um modelo que representa adequadamente as iterações de sinais e os relacionamentos de causa e efeito para o sistema a fim de que o processo de análise possa ser efetuado. Na modelagem teórica, um modelo é obtido através da aplicação de leis conhecidas que regem os eventos essenciais. O modelo matemático que resulta disso é um conjunto de equações relacionando as variáveis representativas do sistema.

#### 4.1.1. Redes Neurais Artificiais

Redes neurais artificiais são sistemas computacionais baseados em ligações onde são usados nós simples ou neurônios para formar uma rede de nós. Essa técnica é inspirada em estruturas do cérebro como os neurônios. Rede Neurais Artificiais (RNA), no campo da inteligência artificial, são sistemas não lineares que imitam o mecanismo de processamento do cérebro humano.

As redes neurais possuem nós ou unidades de processamento, estes nós estão conectados através de pesos. Cada unidade possui links para outras unidades, no qual recebem e enviam sinais. Cada unidade pode possuir memória local. Estas unidades são a simulação dos neurônios, recebendo e retransmitindo informações.

Uma rede neural pode possuir múltiplas camadas. Quanto maior o número de camadas, melhor a capacidade de aprendizado. O número de camadas define a capacidade de representação das relações entre o espaço de entrada e de saída. O

modelo *perceptron* de múltipla camada (MLP) é caracterizado pela existência de uma camada intermediária, além das camadas de saída e entrada. Se houver apenas uma camada intermediária, a MLP pode aproximar com qualquer grau de aproximação qualquer função contínua (Hornik et. al, 1989). Duas ou mais camadas possibilitam aproximação de qualquer função contínua ou não.

Redes neurais artificiais estão muito associadas às sinapses entre neurônios, à adaptação de conexões. Entretanto, existem modelos onde as conexões não são adaptadas, mas, sim, apenas transmitem estimulação entre neurônios. Esses modelos são conhecidos como redes neurais sem pesos. Há também modelos onde as sinapses não são adaptadas, mas calculadas previamente, sendo então a tarefa de otimização.

As aplicações de redes neurais artificiais se baseiam principalmente em sua característica de aprendizado. Atualmente, as aplicações são bastante abrangentes como utilização para *data minning*, OCR (*Optical Character Recognition*), mercado financeiro, sistemas de controle, previsões, reconhecimentos de voz e imagem entre outras.

As redes neurais podem ser classificadas como *feedforward* e *feedback*. O primeiro caso consiste de entradas e saídas apresentadas para treinamento e problemas de instabilidade não ocorrem. As entradas de uma rede deste tipo são conectadas diretamente às saídas para efetuar o processamento em cascata e o treinamento é caracterizado por ser supervisionado. O segundo caso é caracterizado por possuir pelo menos um laço de realimentação. As redes *feedback* são caracterizadas por possuírem um aprendizado não-supervisionado e apresentam problemas de instabilidade.

Para Redes *Feedforward* Multicamadas os nós da camada de entrada servem de entrada para a primeira camada interna e assim por diante até a obtenção da saída. Ainda, pode-se dizer que uma RNA cuja camada interna contém neurônios que efetuam todas as ligações com a camada anterior é chamada de rede completamente conectada.

Existem vários algoritmos de treinamento para uma RNA *feedforward* com arquitetura *backpropagation* como o método do Gradiente e o filtro de Kalman. O método do Gradiente é um método determinístico que se baseia na retro-propagação do erro de camadas subsequentes. Cada vez que o padrão atinge uma camada de saída o erro quadrático é computado e comparado a um erro arbitrado, o erro obtido é utilizado para alterar os pesos sinápticos que ligam as camadas de saída às anteriores. O método do filtro de Kalman é um método estocástico e sua operação se dá pela minimização do erro quadrático médio e parte-se do pressuposto que o sistema deva ser pelo menos linearizável em torno de um ponto nominal. O ruído do sistema é do tipo gaussiano de média nula e a variância constante. Devido às características de retro-propagação é possível o processamento paralelo dos neurônios. O

processamento paralelo permite o uso da retro-propagação e da filtragem de Kalman em nível neural, portanto o processamento e aprendizado da rede podem ser distribuídos paralelamente, diminuindo, assim, a memória utilizada para armazenar a matriz de covariância dos parâmetros.

#### **4.1.2. Estruturas de Integração Numéricas**

Nem todos sistemas de equações diferenciais possuem soluções analíticas, principalmente se tratando de equações do tipo não-linear, neste caso pode-se obter somente soluções numéricas. Integradores numéricos são utilizados para resolver tais sistemas diferenciais. São conhecidos dois métodos principais para resolver um sistema de equações diferenciais de primeira ordem ordinárias, o método de passo simples e o método de múltiplos passos. O método de passo simples utiliza e computa o valor de apenas um passo  $h$  para propagação da solução. Temos como exemplo desse método os integradores do tipo Runge-Kutta de mais alta ordem. Já o método dos múltiplos passos usam os valores da função e suas derivadas para vários pontos anteriores. Alguns exemplos são Adams-Bashforth, Adam-Moulton.

#### **4.1.3. Representação de Sistemas Dinâmicos**

Serão abordadas algumas representações para sistemas dinâmicos com metodologias neurais. São elas: metodologia NARMAX, metodologia das derivadas instantâneas, metodologia das derivadas médias. Como a metodologia das derivadas médias será a utilizada no projeto, sua abordagem será mais aprofundada.

- **Metodologia NARMAX:**

Essa metodologia se baseia na utilização de entradas atrasadas, a solução é propagada por meio de uma técnica que usa informações de estados e tempos anteriores. O passo de integração é fixo e a rede aprende e modela um discretizador da dinâmica do sistema.

- **Metodologia das derivadas instantâneas:**

Esse método utiliza para treinamento da rede a função de derivadas instantâneas. A obtenção dessa função foi proposta por Wang e Lin (1998) e o método se resume na construção destas redes de acordo com o método de aproximação de Runge-Kutta que estima precisamente a mudança nas derivadas instantâneas dos estados do sistema.

- **Metodologia das derivadas médias:**

Essa metodologia utiliza uma rede *feedforward* em conjunto com um



integrador numérico de primeira ordem de Euler, obtendo precisão semelhante àquela obtida por integradores de alta ordem. Nesse método ao invés de utilizar as derivadas instantâneas, são inseridas internamente na estrutura de integração do tipo Euler as funções de derivadas médias. As principais diferenças que surgem são que no método das derivadas médias o passo de integração é fixo e a precisão da modelagem é garantida através da aproximação com que a rede neural é treinada, já o método das derivadas instantâneas o passo de integração pode ser variado para auxiliar no ajuste do erro de propagação do sistema dinâmico em cada passo. No método das derivadas médias consegue-se ter uma elevada precisão com o uso de um método de integração de primeira ordem de Euler, e o método das derivadas instantâneas só consegue ter uma boa precisão se a rede neural for utilizada dentro de um integrador de alta ordem. Essa última diferença traz um fato muito importante de que o método das derivadas médias possuirá expressões analíticas para a retro propagação bem mais simples do que o método das derivadas instantâneas.

#### **4.2. Estruturas de Controle Preditivo Neural**

O controle de sistemas dinâmicos não-lineares utilizando-se de redes neurais com arquitetura *feedforward* foi proposto no final da década de 80 (Hunt et al., 1992). Quando se trata de sistemas lineares, várias técnicas de projetos de sistemas de controles são conhecidas, mas quando se trata de sistemas não-lineares são necessárias várias conjecturas de modo a ter-se um embasamento teórico consistente.

Inicialmente as redes neurais foram trabalhadas empiricamente e consideradas como sendo um método experimental para resolução de sistemas dinâmicos não-lineares. Nas últimas décadas têm surgido avanços em relação aos resultados teóricos, porém os modelos matemáticos envolvidos partem do princípio do conhecimento fenomenológico do sistema a ser modelado. Portanto, quando o problema envolve sistemas não-lineares, sistemas com complexidade elevada a busca de soluções caminha em direção de sistemas heurísticos, em decisões baseadas em lógicas nebulosa, em redes neurais, etc. Estes algoritmos são baseados em dados pré-existentes, onde a memória histórica é a principal fonte de informação.

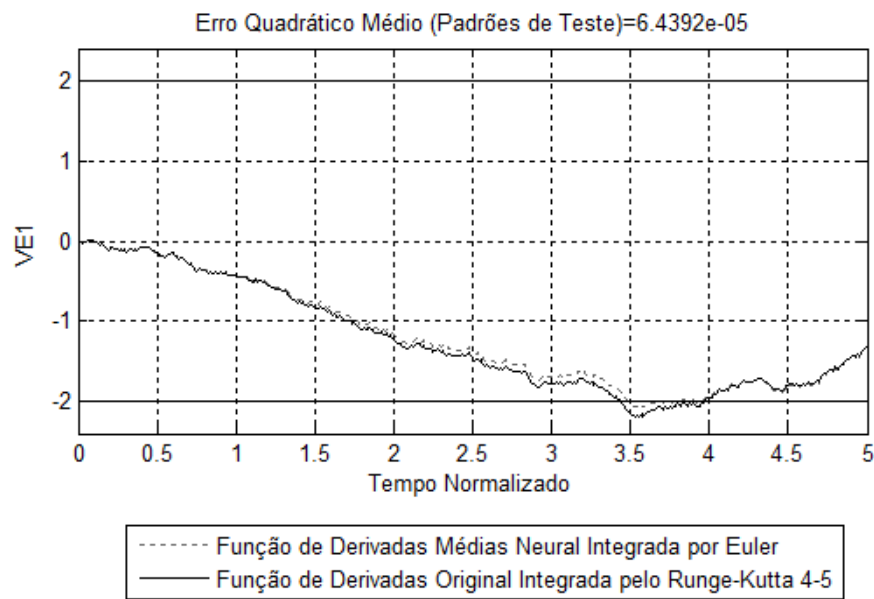
O controle preditivo é uma abordagem adequada para utilizar redes neurais (Phan e Xing, 1995). As etapas para a utilização constituem da identificação do sistema dinâmico, que corresponde ao treinamento de um modelo e da determinação da lei de controle.

## 5. Resultados Obtidos

Ao longo do primeiro semestre foi estudada a teoria necessária pra se colocar o projeto em prática. Para finalizar o período de aprendizado foi realizado um teste no *software* utilizando o método das derivadas médias. O sistema de equação diferencial não-linear de primeira ordem utilizado para realizar a experiência foi:

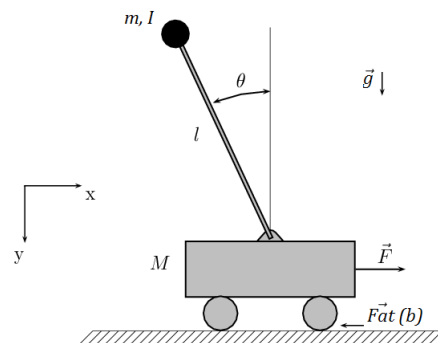
$$\begin{cases} \dot{y}_1 = \sin(y_1 \cdot y_2) + u \\ \dot{y}_2 = (y_1^2 + y_2^2) \cdot u \end{cases}$$

Sendo  $y_1$  e  $y_2$  as variáveis de estado e  $u$  a variável de controle. O teste foi submetido a 48 horas de processamento e foi obtido um erro quadrático médio de  $6.4392e - 05$ . A seguir um gráfico comparativo do resultado:



O objetivo do projeto é o controle do pêndulo invertido, entretanto aplicando a segunda Lei de Newton para o movimento do pêndulo, obtemos um sistema de equações diferenciais ordinárias não-linear de segunda ordem:

$$\begin{cases} (M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = F \\ (I + ml^2)\ddot{\theta} + mgl \sin \theta = -ml\ddot{x} \cos \theta \end{cases}$$



Com um pouco de manipulação matemática conseguimos transformar esse sistema anterior em um sistema de equações ordinárias não-linear de primeira ordem. Esse novo sistema é o que aplicaremos no *software*. Sendo  $x = x_1$ ,  $\dot{x} = x_2$ ,  $\theta = x_3$ ,  $\dot{\theta} = x_4$  e  $u$  a posição do centro do carrinho em relação a origem do sistema cartesiano, a velocidade horizontal do carrinho, o ângulo que a haste do pêndulo faz com a vertical, a velocidade angular da haste e a força externa aplicada ao carrinho respectivamente. Temos o seguinte sistema:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{[u - bx_2 + mlx_4^2 \sin(x_3)](I + ml^2) + (ml)^2 g \sin(x_3) \cos(x_3)}{(I + ml^2)(M + m) - [ml \cos(x_3)]^2} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{\{[u - bx_2 + mlx_4^2 \sin(x_3)] \cos(x_3) + (M + m)g \sin(x_3)\}ml}{[ml \cos(x_3)]^2 - (I + ml^2)(M + m)} \end{cases}$$

Esse sistema é de primeira ordem e pode ser aplicado no *software*. A demonstração para encontrar o novo sistema está no Anexo A.

## 6. Conclusões

O projeto até então ratificou o método de utilização das derivadas médias. Foram feitas explicações a respeito da Representação de Sistemas Dinâmicos de Redes Neurais Artificiais de um modo superficial, porém bem explicativo. É esperado que o *software* funcione bem com o sistema de equações diferenciais ordinárias de primeira ordem exposto na seção 4.

## 7. Plano de Trabalho e Cronograma das Etapas Seguintes

O objetivo do projeto é aplicar redes neurais na resolução de equações diferenciais ordinárias não lineares e sua posterior aplicação em teoria de controle preditivo para resolver o problema do pêndulo invertido. No primeiro semestre o plano era adquirir o conhecimento teórico necessário para a execução do projeto. Considera-se que o objetivo do primeiro semestre foi concluído com êxito. Para o segundo semestre, com as equações do movimento do pêndulo invertido já transformadas em um sistema de primeira ordem, o objetivo torna-se aplicar o conhecimento adquirido no primeiro semestre para resolver o problema do pêndulo invertido e assim obter conclusões a respeito do assunto.

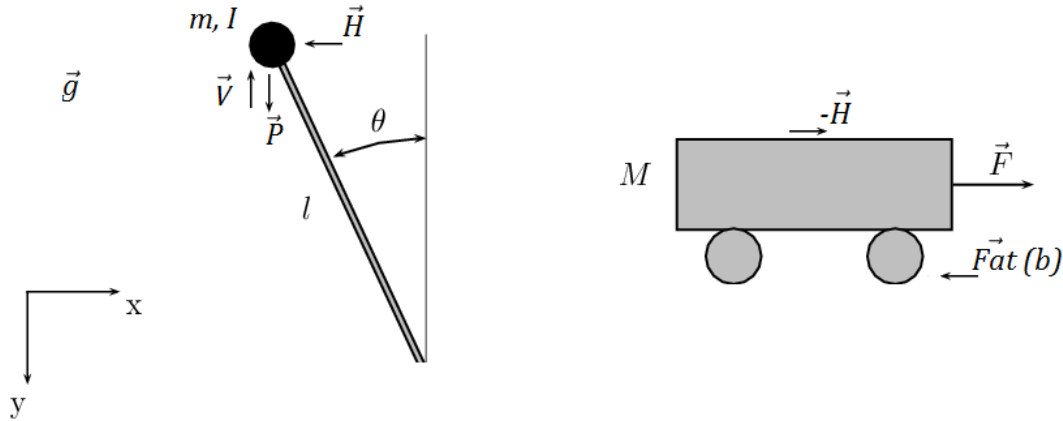
## 8. Bibliografia

- Cotter, N. E. (1990). The Stone-Weierstrass and its application to neural networks. *IEEE Transactions on Neural Networks*. v. 1, n. 4, pp. 290-295.
- Haykin, S. (1999). *Neural networks: a comprehensive foundation*. 2nd ed., New Jersey: Prentice-Hall, Inc..
- Hornik, K., Stinchcombe, M. and White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, n. 5, pp. 359-366.
- Hunt, J. K., Sbarbaro, D., Zbikowski, R. and Gawthrop, P. (1992). Neural networks for control system-a survey, *Automatica* **28**(6): 1083–1112.
- Jang, J.-S. R., Sun, C.-T and Mizutani, E. (1997). *Neuro-Fuzzy and soft computing: a computational approach to learning and machine intelligence*. Prentice-Hall, Inc.
- Melo, R. P. (2008). *Metodologia de modelagem empírica utilizando integradores neurais aplicada a sistemas dinâmicos não-lineares*. 136f. Tese de Mestrado. Instituto Tecnológico de Aeronáutica, São José dos Campos.
- Melo, R. P. and Tasinaffo, P. M. (2010). Uma metodologia de modelagem empírica utilizando o integrador neural de múltiplos passos do tipo Adams-Bashforth. *SBA. Sociedade Brasileira de Automática*, v. 21, n. 5, pp. 487-509.
- Mitchell, M. (1998). *An Introduction to genetic algorithms*. First MIT Press paperback edition.
- Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks, *IEEE Transactions on Neural Networks* 1: 4–27.
- Narendra, K. S. (1996). Neural networks for control, *Proceedings of the IEEE* 84: 1385–1406.
- Rios Neto, A. (2001). Dynamic systems numerical integrators in neural control schemes. *V Congresso Brasileiro de Redes Neurais*, Rio de Janeiro-RJ, pp. 85–88.
- Russell, S. and Norvig, P. (2003). *Artificial Intelligence*. 2nd ed., New Jersey: Prentice-Hall, Inc.
- Spooner, J. T., Maggiore, M., Ordóñez, R. and Passino, K. M. (2002). *Stable adaptive control and estimation for nonlinear systems neural and fuzzy approximator techniques*. New York: Wiley-Interscience.
- Tasinaffo, P. M. (2003). *Estruturas de integração neural feedforward testadas em problemas de controle preditivo*. 230 p. INPE-10475-TDI/945. Doctoral Thesis, Instituto Nacional de Pesquisas Espaciais -INPE, São José dos Campos, Brasil.
- Tasinaffo, P. M. and Rios Neto, A. (2005). Mean derivatives based neural euler integrator for nonlinear dynamic systems modeling. *Learning and Nonlinear Models*, v. 3, pp. 98–109.
- Tasinaffo, P. M. and Rios Neto, A (2006). Predictive control with mean derivative based neural euler integrator dynamic model. *SBA. Sociedade Brasileira de Automática*, v. 18, n. 1, pp. 94-105.
- Wang, Y.-J. and Lin, C.-T. (1998). Runge-Kutta neural network for identification of dynamical systems in high accuracy. *IEEE Transactions On Neural Networks*, v. 9, n. 2, p. 294-307.
- Oliveira, D. G.; Tasinaffo, P. M. Controle Preditivo com Rede Neural Artificial Aplicado ao Problema de Equilíbrio do Pêndulo Invertido Não-Linear. Santos, N.18, ano 5.
- Tasinaffo, P. M.; Neto, A. R. Modelagem de Sistemas Dinâmicos com Integrador Neural de Euler Baseado em Derivadas Médias.

## 9. Anexo A

Segue abaixo a demonstração de como encontrar o sistema de equações diferenciais ordinárias não-lineares de primeira ordem exposto na seção 4.

Sendo  $H$ ,  $V$  as forças horizontal e vertical respectivamente que a barra aplica na bola, temos o seguinte diagrama de forças para o sistema:



Aplicando a Segunda Lei de Newton para o movimento horizontal do carrinho e da bola, temos as equações:

$$\begin{cases} F - H - b\dot{x} = M\ddot{x} & (a) \\ H = m \frac{d^2}{dt^2} (x + l \sin \theta) = m\ddot{x} + ml\ddot{\theta} \cos \theta + ml\dot{\theta}^2 \sin \theta & (b) \end{cases}$$

Substituindo (b) em (a), temos:

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

No referencial do carrinho, a bola faz um movimento de rotação em torno do ponto de fixação entre a barra e o carrinho. Fazendo as devidas alterações por ser um referencial com aceleração não nula, temos a seguinte equação:

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

De (c) e (d) obtemos o seguinte sistema de equações:

$$\begin{cases} (M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = F & (1) \\ (I + ml^2)\ddot{\theta} = -mgl \sin \theta - ml\ddot{x} \cos \theta & (2) \end{cases}$$

Faça  $x_1 = x$ ,  $x_2 = \dot{x}$ ,  $x_3 = \theta$ ,  $x_4 = \dot{\theta}$ . Isole  $\ddot{\theta}$  em (2), temos:

$$\ddot{\theta} = -\frac{ml \cos \theta \ddot{x}}{(I + ml^2)} - \frac{mgl \sin \theta}{(I + ml^2)} \quad (3)$$

Agora substitua (3) em (1) e isole  $\ddot{x} = \dot{x}_2$ :

$$\dot{x}_2 = \frac{[u - bx_2 + mlx_4^2 \sin(x_3)](I + ml^2) + (ml)^2 g \sin(x_3) \cos(x_3)}{(I + ml^2)(M + m) - [ml \cos(x_3)]^2} \quad (4)$$

Isole  $\ddot{x}$  em (2), temos:

$$\ddot{x} = -\frac{(I + ml^2)\ddot{\theta}}{ml \cos \theta} - g \tan \theta \quad (5)$$

Substituindo (5) em (1) e isolando  $\ddot{\theta} = \dot{x}_4$ , temos:

$$\dot{x}_4 = \frac{\{[u - bx_2 + mlx_4^2 \sin(x_3)] \cos(x_3) + (M + m)g \sin(x_3)\}ml}{[ml \cos(x_3)]^2 - (I + ml^2)(M + m)} \quad (6)$$

Sabendo que  $\dot{x}_1 = x_2$ ,  $\dot{x}_3 = x_4$  e com (4) e (6), temos o seguinte sistema de equações diferenciais ordinárias não lineares de primeira ordem:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \frac{[u - bx_2 + mlx_4^2 \sin(x_3)](I + ml^2) + (ml)^2 g \sin(x_3) \cos(x_3)}{(I + ml^2)(M + m) - [ml \cos(x_3)]^2} \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{\{[u - bx_2 + mlx_4^2 \sin(x_3)] \cos(x_3) + (M + m)g \sin(x_3)\}ml}{[ml \cos(x_3)]^2 - (I + ml^2)(M + m)} \end{cases}$$

Como queríamos demonstrar.