

ANÁLISE DAS INTERFACES DOS SIMULADORES DIGITAIS E SUAS APLICAÇÕES

Alisson Moisés Moreira de Souza¹, Ana Cristina C. Lyra²

Resumo — *A simulação digital mostra-se como importante ferramenta para desenvolvimentos e para aprendizagem. Trata-se de análise de diferentes interfaces homem-máquina usadas nos simuladores digitais e suas aplicações. É simulada a partida da máquina de indução. São consideradas três ambientes de simulação: o "prompt" de comandos do Mat Lab; o Simulink usando diagramas de simulação analógica; e modelos padronizados na biblioteca Power System, MatLab 6.12/Simulink*

O ambiente de "baixo nível" (prompt) exige familiaridade com modelos matemáticos, indicado ao ensino de modelagem e integração numérica, ou ao desenvolvimento de modelos inéditos, exigindo poucos recursos computacionais. A linguagem de simulação analógica permite estruturar e personalizar os modelos, o que é útil no processo pedagógico. Finalmente, o uso de modelos padronizados (Tool Box) exige pouca habilidade com modelagem, substituída pela familiaridade com a implementação de bancada, aplicando-se, geralmente, à exploração de propriedades macroscópicas dos modelos. Por outro lado, exige abundantes recursos computacionais.

Índice—Simulação digital, interfaces homem-máquina, máquina de indução, Mat Lab

INTRODUÇÃO

O uso de simuladores digitais é hoje uma realidade para os engenheiros. Seja na construção de máquinas elétricas, na indústria automobilística, nas faculdades e até na economia, a simulação digital mostra-se como importante ferramenta nos processos de desenvolvimento e de aprendizagem. Torna-se possível prever situações indesejadas, antecipar decisões, dimensionar parâmetros ou, simplesmente, explorar propriedades dos sistemas em simulação, garantindo economia de tempo e dinheiro.

Analisam-se três diferentes interfaces homem-máquina usadas nos simuladores digitais e

suas aplicações na engenharia, em especial na elétrica.

A história dos simuladores digitais quase se mistura com a história da programação. A simulação digital apareceu dentro das grandes universidades como ferramenta aplicada ao estudo de sistemas físicos ou matemáticos, para explorar propriedades e validar teorias, geralmente em situações associadas à pesquisa ou ao desenvolvimento, que exigiam número elevado de cálculos, inviáveis até então. Com poucos, e caros, recursos computacionais, a simulação digital possuiu uma infância bem difícil, com interfaces bastante desconfortáveis ao usuário, próximas ao nível de abstração da máquina. Rapidamente, grandes laboratórios e grandes empresas vislumbraram o potencial da simulação digital para o desenvolvimento e para a pesquisa.

Com o desenvolvimento das linguagens de programação e a expansão dos recursos computacionais, a simulação digital também se desenvolveu, aumentando a proximidade com o usuário e o grau de especialização.

Hoje, com o mercado de software consolidado, existem inúmeros simuladores digitais voltados a cada área específica, em diferentes níveis de proximidade com o usuário e de acordo com cada aplicação.

METODOLOGIA

A metodologia adotada foi a simulação do modelo matemático da máquina elétrica de indução trifásica em diferentes interfaces de simulação em três níveis de proximidade com o usuário, usando, para isto, o software MatLab 6.12 e seus pacotes.

A primeira interface, ou linguagem, de baixo nível de proximidade com o usuário, refere-se ao "prompt de comandos" do MatLab 6.12. A segunda interface refere-se a diagramas de simulação analógica usando o pacote *Simulink*, também no MatLab 6.12, em ambiente gráfico.

A última interface analisada, de alto nível de proximidade com o usuário, refere-se ao Toolbox

¹ Alisson Moisés Moreira de Souza, Mestrando em Engenharia Elétrica, na Faculdade de Engenharia Elétrica da UNICAMP, alissonmail2001@yahoo.com fone (015)-2433243

² Ana Cristina C. Lyra, Professora Doutora na Faculdade de Engenharia Elétrica, na UNICAMP, Rua Albert Einstein 400, UNICAMP, Barão Geraldo, Campinas, São Paulo, Brasil, aclyra@fee.unicamp.br

da biblioteca Power System, também no *Simulink*, MatLab 6.12, onde a modelagem é implementada em ambiente gráfico com modelos padronizados ou pré-formatados.

MODELAMENTO MATEMÁTICO

O modelo matemático utilizado é o modelo da máquina de indução trifásica alimentada via estator, escrito na referência síncrona através de eixos ortogonais conhecidos por dq. As simulações consideram uma máquina de indução de 3hp.

O conjunto de equações diferenciais que sintetiza o modelo é listado a seguir. As equações de (1) a (4) são as equações elétricas de estator e rotor, compostas pelas parcelas ôhmicas e indutivas nos enrolamentos.

A equação (5) representa o acoplamento entre as correntes e os fluxos magnéticos da máquina de indução. Finalmente, as equações (6) e (7) representam o torque eletromagnético gerado, juntamente com acoplamento mecânico do momento de inércia e a carga mecânica aplicadas ao eixo.

Embora haja sete equações, matematicamente, tem-se um sistema diferencial não-linear de 5ª ordem: duas equações de estator, duas equações de rotor e uma equação mecânica.

$$\text{estator} \left\{ \begin{array}{l} v_{qs} = \frac{1}{\omega_b} \cdot \frac{d\varphi_{qs}}{dt} + \frac{\omega}{\omega_b} \cdot \varphi_{ds} + r_s \cdot i_{qs} \quad (1) \\ v_{ds} = \frac{1}{\omega_b} \cdot \frac{d\varphi_{ds}}{dt} - \frac{\omega}{\omega_b} \cdot \varphi_{qs} + r_s \cdot i_{ds} \quad (2) \end{array} \right.$$

$$\text{rotor} \left\{ \begin{array}{l} v_{qr} = \frac{1}{\omega_b} \cdot \frac{d\varphi_{qr}}{dt} + \frac{(\omega - \omega_r)}{\omega_b} \cdot \varphi_{dr} + r_r \cdot i_{qr} \quad (3) \\ v_{dr} = \frac{1}{\omega_b} \cdot \frac{d\varphi_{dr}}{dt} - \frac{(\omega - \omega_r)}{\omega_b} \cdot \varphi_{qr} + r_r \cdot i_{dr} \quad (4) \end{array} \right.$$

Matrizfluxo-corrente

$$\begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{qr} \\ i_{dr} \end{bmatrix} = \begin{bmatrix} a & 0 & b & 0 \\ 0 & a & 0 & b \\ b & 0 & a & 0 \\ 0 & b & 0 & a \end{bmatrix} \cdot \begin{bmatrix} \varphi_{qs} \\ \varphi_{ds} \\ \varphi_{qr} \\ \varphi_{dr} \end{bmatrix} \quad (5)$$

$$\left\{ \begin{array}{l} \text{Equação do Torque Eletromagnético} \\ \tau_{ele} = \frac{3}{2} \cdot \frac{X_m}{\omega_b} \cdot (i_{qs} \cdot i_{dr} - i_{ds} \cdot i_{qr}) \quad (6) \\ \text{Equação mecânica} \\ \frac{d\omega_r}{dt} = \frac{1}{J} (\tau_{ele} - T_c) \quad (7) \end{array} \right.$$

LEGENDA DAS VARIÁVEIS DAS EQUAÇÕES

v_{qs}	tensão do eixo q do estator;
v_{ds}	tensão do eixo d do estator;
v_{qr}^*	tensão eixo q de rotor refletida no estator;
v_{dr}^*	tensão eixo d de rotor refletida no estator;
r_{qs}	resistência do eixo q do estator;
r_{ds}	resistência do eixo d do estator;
r_{qr}^*	resistência eixo q de rotor refletida no estator;
r_{dr}^*	resistência eixo d de rotor refletida no estator;
i_{qr}^*	corrente eixo q rotor refletida no estator;
i_{dr}^*	corrente eixo d rotor refletida no estator;
i_{qs}	corrente do eixo q do estator;
i_{ds}	corrente do eixo d do estator;
Ψ_{qs}	fluxo por segundo do eixo q do estator;
Ψ_{ds}	fluxo por segundo do eixo d do estator;
Ψ_{qr}^*	fluxo por segundo do eixo q do rotor refletido no estator;
Ψ_{dr}^*	fluxo por segundo do eixo d do rotor refletido no estator;
ω_b	frequência base da rede de alimentação do estator;
ω	frequência da referência genérica utilizada no modelo, no caso a frequência síncrona;
ω_r	frequência mecânica do rotor;
J	momento de inércia do eixo do rotor;
T_c	torque de carga.
X_m	reatância mútua entre estator e rotor;

$$b = \frac{X_m^3 - X_{lm}^2 \cdot X_m}{X_{lm}^4 - 2 \cdot X_{lm}^2 \cdot X_m + X_m^4}$$

$$a = \frac{X_{lm}^3 - X_{lm} \cdot X_m^2}{X_{lm}^4 - 2 \cdot X_{lm}^2 \cdot X_m + X_m^4}$$

O modelo matemático é aplicado às diferentes interfaces de simulação. A variável de saída analisada é a velocidade durante a partida.

INTERFACE DE BAIXO NÍVEL: “PROMPT” DE COMANDOS DO MATLAB 6.12

Nesta interface, a programação do modelo é feita reescrevendo as equações diferenciais em forma de equações de estado, para permitir a utilização de métodos numéricos de integração. O método numérico de integração utilizado é o método de Euler Modificado.

A linguagem de programação é composta de uma seqüência de comandos estruturados no “prompt” do MatLab 6.12. Trata-se de linguagem semelhante à linguagem C, permitindo estruturas de programação do tipo *for*, *if*, *while*, entre outras.

Em primeiro lugar definem-se os parâmetros da máquina elétrica: resistências, indutâncias, momento de inércia, carga nominal, frequência base, etc. A seguir, inicializam-se as variáveis do programa, como as correntes, a velocidade, torque, fluxos magnéticos, tempo e outras variáveis auxiliares.

Então, entra-se no loop das iterações, uma estrutura do tipo *for*, começando com $t=0$ até o tempo final de simulação, onde o sistema diferencial é resolvido passo a passo.

Dentro do *for*, calcula-se cada derivada das variáveis de estado, no caso, os fluxos magnéticos e a velocidade. Então, integra-se cada variável de estado, usando uma aproximação linear (Euler Modificado). A seguir calculam-se as correntes e, a partir destas, o torque. Finalmente, consegue-se a velocidade. Isto é feito para cada incremento da variável tempo, cujo passo é 1ms. A cada passo, a velocidade e outras variáveis de interesse (torque, tensões e correntes) são armazenadas em estruturas do tipo vetor de dados, conhecidas como *Array*.

Segue, abaixo, o conjunto de comandos responsável pela simulação do modelo, nesta interface.

INTERFACE DE BAXIO NÍVEL DE PROXIMIDADE COM O USUÁRIO

```
%%Interface de baixo nível de proximidade com o usuário
clear all
%%Parametros-simulação
passo=0.001;
tfinal=5;
wref=377;
%%Parametros-máquina
wb=377;
rs=0.435;
rr=0.816;
xm=26.13;
xls=0.754;
xlm=xm+xls;
J=0.089;
a=(xlm^3-xlm*xm^2)/(xlm^4-2*(xlm*xm)^2+xm^4)
b=(-xm*xlm^2+xm^3)/(xlm^4-2*(xlm*xm)^2+xm^4)
vsmax=1.22*220;
vrotor=1.22*0;
tc=11.9;
%%INICIALIZACAO
fiqs=0;
fids=0;
fidr=0;
fiqr=0;
ids=0;
iqs=0;
idr=0;
iqr=0;
wr=0;
triangular=0;
%% Loop
for t=0:passo:tfinal;
%%Define vds, vqs,vdr,vqr e tcarga;
vds=vsmax*cos((wb-wref)*t);
vqs=vsmax*sin((wb-wref)*t);
%
vdr=0;
vqr=0;

if t<3
    tcarga=0;
elseif t>4
    tcarga=2*tc;
```

```
else t>1
    tcarga=tc;
end;
%
%%DERIVADAS
dfiqs_dt=vqs*wb-wref*fidse-rs*iqse*wb;
dffids_dt=vds*wb+wref*fiqse-rs*idse*wb;
dffiqr_dt=vqr*wb-(wref-wre)*fidre-rr*iqre*wb;
dffidr_dt=vdr*wb+(wref-wre)*fiqre-rr*idre*wb;
%%FLUXOS
fiqs=fiqs+(dfiqs_dte+dfiqs_dt)/2*passo;
fids=fids+(dffids_dte+dffids_dt)/2*passo;
fiqr=fiqr+(dffiqr_dte+dffiqr_dt)/2*passo;
fidr=fidr+(dffidr_dte+dffidr_dt)/2*passo;
%%CORRENTES e TORQUE ELETROMANETICO
iqs=a*fiqs+b*fiqr;
ids=a*fids+b*fidr;
iqr=b*fiqs+a*fiqr;
idr=b*fids+a*fidr;
tele=3/2*xm/wb*(iqs*idr-ids*iqr)-tcarga;
dwr_dt=1/J*(tele);
wr=wr+(dwr_dte+dwr_dt)/2*passo;
%
plot(tempo,velo)

velo(t/passo+1)=wr;
tempo(t/passo+1)=t;
tele_(t/passo+1)=tele;
ids_(t/passo+1)=ids;
iqs_(t/passo+1)=iqs;
is_(t/passo+1)=sqrt(ids^2+iqs^2);
triangular_(t/passo+1)=triangular;
vdr_(t/passo+1)=vdr;
vqr_(t/passo+1)=vqr;
end;
```

INTERFACE : DIAGRAMAS DE SIMULAÇÃO ANALÓGICA (MATLAB 6.12/SIMULINK)

Este tipo de interface foi muito consagrado entre as aplicações de controle de sistemas, dada sua proximidade com a implementação dos sistemas de controle através de elementos analógicos e sua capacidade de sintetizar propriedades do sistema de maneira visual.

Nesta interface também é preciso reescrever as equações diferenciais na forma de equações de estado. A seguir iniciou-se a construção dos diagramas de simulação analógica a partir dos elementos do pacote Matlab 6.12/Simulink. São utilizados blocos que representam as operações matemáticas das equações de forma visual. Nestes diagramas, pode-se representar operações lineares como soma, ganho, integração, diferenciação; operações não-lineares como o produto entre duas variáveis de estado; ou mesmo os termos não-homogêneos das equações como fontes e o torque de carga, através de entradas senoidais ou degraus no domínio do tempo.

Cada equação de estado é montada e transformada em um bloco fechado, formando um sub-diagrama. Existem quatro blocos, um para as equações de estator, um para as equações de rotor,

um para a equação mecânica e um para matriz corrente-fluxo. Pode-se visualizar a seguir o bloco da equação mecânica.

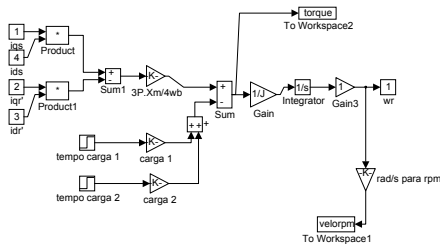


FIGURA.1
EQUAÇÃO MECÂNICA

Então, cria-se um modelo de simulação estruturado e sintetizado através do agrupamento destes sub-diagramas, formando o diagrama de simulação principal, o produto final.

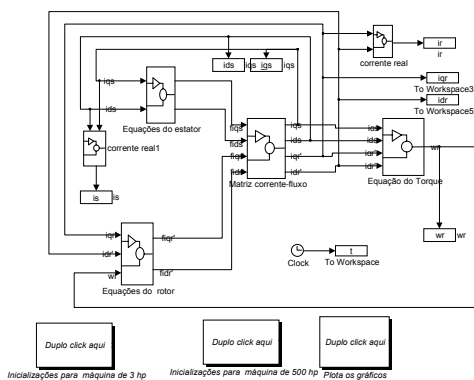


FIGURA.2
DIAGRAMA DE SIMULAÇÃO PRINCIPAL

INTERFACE DE ALTO NÍVEL : BLOCOS PRÉ-FORMATADOS (BIBLIOTECA POWER SYSTEM, MATLAB 6.12/SIMULINK)

Este tipo de interface, de alto nível de proximidade com o usuário, torna o processamento praticamente transparente e, portanto, não é preciso reescrever as equações. Na verdade, neste caso, não é preciso nem mesmo conhecê-las, já que o modelo da máquina de indução é padronizado pelo software. A construção do modelo é iniciada utilizando os elementos oferecidos pela biblioteca Power System do Simulink. Os blocos utilizados são os elementos que seriam utilizados para implementação prática em um laboratório: a máquina de indução, uma fonte trifásica, a carga mecânica e, finalmente, o

osciloscópio.

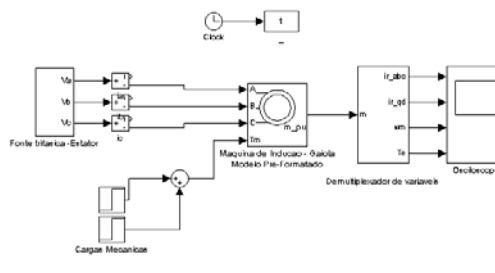


FIGURA.3
DIAGRAMA DE SIMULAÇÃO DE ALTO NÍVEL

RESULTADOS

A seguir, o modelo da máquina de indução trifásica é simulado a partir das três interfaces, fornecendo, como variável de análise, a velocidade. É simulada a partida da máquina nas seguintes condições: a máquina, de 3hp, é partida em vazio, nas condições de tensão e frequência nominais até que a velocidade de regime seja atingida. Então, é adicionada ao eixo carga mecânica de valor nominal, aguardando a nova velocidade de regime. Finalmente, adiciona-se uma segunda carga mecânica, também de valor nominal. Como havia de se esperar, todos os modelos forneceram os mesmos resultados de saída.

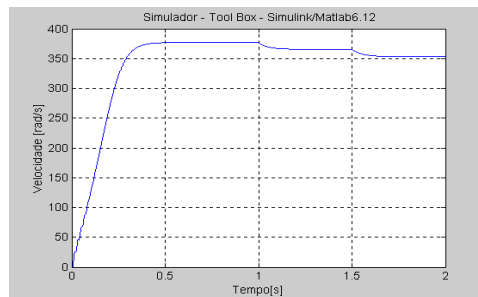


FIGURA.4
SIMULAÇÃO 1 – ALTO NÍVEL

FIGURA.5
SIMULAÇÃO 2 – DIAGRAMA SIM.ANALÓGICA



FIGURA.6
SIMULAÇÃO 3 - BAIXO NÍVEL

CONCLUSÕES

Conclui-se que o ambiente de “baixo nível” (prompt do MatLab) exige familiaridade com os modelos matemáticos, sendo indicado ao ensino de modelagem matemática e métodos numéricos de integração. Aplica-se, ainda, à pesquisa de modelos inéditos, onde a padronização não é bem vinda.

A linguagem de simulação analógica exige também habilidades com os modelos matemáticos, permitindo personalizá-los e estruturá-los, o que é útil no processo pedagógico.

Já o uso de simuladores de alto nível de proximidade com o usuário (no caso, Tool Box, MatLab 6.12/Simulink) exige pouca habilidade com modelagem, que deve ser substituída pela familiaridade com a implementação de bancada e aplica-se, geralmente, à pesquisa com elementos clássicos, disponíveis no programa.

Aplica-se também à exploração de propriedades macroscópicas dos modelos. Por outro lado, exige abundantes recursos computacionais e permite pouco domínio do processamento dos modelos em simulação.

Nota-se grande tendência ao uso de simuladores de alto nível, especialmente em aplicações comerciais e nos processos produtivos, onde a padronização e o tempo de desenvolvimento são fatores cruciais. Por outro lado, a padronização aplicada à modelagem pode prejudicar o processo criativo aplicado ao desenvolvimento científico, na medida em que não permite mudanças matemáticas estruturais, presentes nos simuladores digitais de baixo nível.

Deste modo, o uso de simuladores digitais no processo pedagógico deve ponderar o nível de proximidade com o usuário aplicado ao modelamento a fim de permitir uma formação ampla, preparando fundamentos necessários aos futuros cientistas, importantes para o desenvolvimento científico e, ao mesmo tempo, profissionais que serão absorvidos pelo mercado produtivo.

BIBLIOGRAFIA

- [1] Ong, Chee-Mun, DYNAMIC SIMULATION OF ELECTRIC MACHINERY USING MATLAB/SIMULINK. Editora PRENTICE HALL. <http://www.prenhall.com>
- [2] Krause, P.C., ANALYSIS OF ELECTRIC MACHINERY, editora McGraw-Hill, Singapore, 1987

TABELA 1
SÍNTESE DAS ANÁLISES

Nível de proximidade com o usuário	Baixo nível	Médio nível	Alto nível
Interface	Prompt de comandos	Diagramas de Simulação analógica	Modelos pré-formatados
Linguagem	Próximo à linguagem C	Simulação Analógica (visual)	Blocos prontos (visual)
Modelos Matemáticos	Equações de estado	Blocos e sub-blocos estruturados	Modelos invisíveis ao usuário
Hardware suporte	Pouca capacidade de processamento Plataforma usada: 486 – 66Mhz	Capacidade razoável: Plataforma usada: Pentium 233 Mhz	Grande capacidade de processamento: Plataforma usada: Duron 1100 Mhz
Construção do Modelo	Demorada; trabalhosa; Permite domínio do processamento interno;	Permite estruturação dos blocos; Pouco domínio do processamento interno;	Fácil; Prática; Pouco domínio do processamento interno;
Familiaridades exigidas	Modelagem; programação;	Simulação Analógica;	Implementação de bancada;
Interpretação dos resultados	Depende da implementação;	Depende da implementação;	Padronizada;
Finalidade Principal	Estudos teóricos de modelagem, métodos de integração; Pesquisa com modelos inéditos;	Ensino; Pesquisa;	Estudo de propriedades macroscópicas dos modelos; Exploração de modelos já bem consolidados na literatura; Laboratórios virtuais; Ensino à distancia;