

**UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL**

**ESCOLA DE ENGENHARIA**

**DEPARTAMENTO DE ENGENHARIA ELÉTRICA**

***RODRIGO DA COSTA DUARTE***

**INTERFACE GRÁFICA PARA PROJETO DE CONTROLADOR DE *UPS***

**Porto Alegre**

**2015**

**RODRIGO DA COSTA DUARTE**

**INTERFACE GRÁFICA PARA PROJETO DE CONTROLADOR DE UPS**

*Projeto de Diplomação apresentado ao Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul, como requisito parcial para obtenção de grau Bacharel em Engenharia Elétrica*

**Orientador: Professor Doutor Luís Fernando Alves Pereira**

**Porto Alegre**

**2015**

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Carlos Alexandre Netto

Vice-Reitor: Rui Vicente Oppermann

ESCOLA DE ENGENHARIA

Diretor: Luiz Carlos Pinto da Silva Filho

Vice-Diretora: Carla Schwengber Ten Caten

DEPARTAMENTO DE ENGENHARIA ELÉTRICA

Chefe: Aly Ferreira Flores Filho

Chefe Substituto: Ramon Carlos Poisl

da Costa Duarte, Rodrigo  
INTERFACE GRÁFICA PARA PROJETO DE CONTROLADOR DE  
UPS / Rodrigo da Costa Duarte. -- 2015.  
72 f.

Orientador: Luís Fernando Alves Pereira.

Trabalho de conclusão de curso (Graduação) --  
Universidade Federal do Rio Grande do Sul, Escola de  
Engenharia, Curso de Engenharia Elétrica, Porto  
Alegre, BR-RS, 2015.

1. Interface gráfica de usuário. 2. Fontes  
ininterruptas de energia. 3. Controlador ressonante.  
4. Nobreak. 5. Uninterruptible power supply. I.  
Fernando Alves Pereira, Luís, orient. II. Título.

**RODRIGO DA COSTA DUARTE**

**INTERFACE GRÁFICA PARA PROJETO DE CONTROLADOR DE *UPS***

Este trabalho de conclusão de curso foi analisado e julgado adequado para obtenção de grau Bacharel em Engenharia Elétrica e aprovado em sua forma final pelo Orientador e pela Banca Examinadora designada pelo Departamento de Engenharia Elétrica da Universidade Federal do Rio Grande do Sul

---

**Orientador: Prof. Dr. Luís Fernando Alves Pereira**

---

**Chefe: Prof. Dr. Ály Flores Ferreira Filho**

**Aprovado em: \_\_\_\_ / \_\_\_\_ / \_\_\_\_\_**

**BANCA EXAMINADORA**

---

**Prof. Dr. Luís Fernando Alves Pereira – UFRGS**  
**Doutor pelo Instituto Tecnológico de Aeronáutica, Brasil**

---

**Prof. Dr. Jeferson Vieira Flores – UFRGS**  
**Doutor pela Universidade Federal do Rio Grande do Sul, Brasil**

---

**Prof. Dr. Marcelo Götz – UFRGS**  
**Doutor pela Universidade Federal do Rio Grande do Sul, Brasil**

## **AGRADECIMENTOS**

Primeiramente, agradeço aos meus pais, Rogério Canabarro Duarte e Ione Santos da Costa, por todo o suporte oferecido, não só durante essa etapa da vida, mas durante toda ela. Agradeço também à minha irmã, Rosane da Costa Duarte, por ser uma pessoa presente em minha vida, sempre disposta a me ouvir.

Ao professor Luís Fernando Alves Pereira, pela orientação e tempo gasto durante o desenvolvimento do trabalho.

Aos professores da Escola de Engenharia da Universidade Federal do Rio Grande do Sul e, especialmente, aos integrantes do Departamento de Engenharia Elétrica, pela dedicação à formação de seus alunos.

Um agradecimento especial ao professor Jeferson Vieira Flores, pela disponibilidade em diversos momentos deste trabalho, bem como ao aluno de doutorado Charles Lorenzini e ao aluno de mestrado Bruno Henz Mossmann, sempre dispostos a ajudar.

Finalmente, um grande agradecimento a todos meus amigos, sempre compreendendo minha ausência ao longo desses anos.

## RESUMO

Este trabalho apresenta uma solução em interface gráfica de usuário para o cálculo dos ganhos de um controlador baseado em compensadores ressonantes, aplicado à *UPS* (do inglês, *Uninterruptible Power Supplies*, ou Fontes Ininterruptas de energia). A necessidade do uso de compensadores ressonantes se dá pelo fato que abordagens clássicas, tal qual PID (Proporcional-Integral-Derivativo), não são adequadas para seguimento de referência senoidal. A partir de um *script* em MatLab®, capaz de resolver os problemas necessários para se obter os ganhos dos compensadores ressonantes, será implementada a interface gráfica, baseada nos recursos do *GUI-Guide* (Sigla para *Graphical User Interface – Guide*), aplicação também em MatLab®. Primeiramente, são apresentados ao leitor os blocos disponíveis dentro do editor de interfaces gráficas, e em seguida, é proposta uma interface capaz de capturar os dados de entrada e imprimir na tela os ganhos para o controlador ressonante. A interface oferece a possibilidade de executar uma simulação da *UPS*. Ao final do trabalho, é apresentado resultados de ensaio prático na *UPS*, com o uso dos ganhos obtidos através da interface.

Palavras-chave: Interface gráfica de usuário. Controlador ressonante. Fontes ininterruptas de energia.

## ABSTRACT

This work presents a graphical user interface solution for the computation of the resonant controller parameters applied to Uninterruptible Power Supplies. The requirement to apply resonant compensators is due to the bad behaviour of classical solutions, such as PID controllers, which have poor performance in tracking sinusoidal reference signals. From a MatLab® Script able to solve the problems in finding the individual gains of resonant compensators, this work seeks to develop a MatLab® *GUI*-Guide based graphical interface for this purpose. Firstly, it is presented the available features in the graphical interface's editor, and then, the interface for obtaining the gains of the resonant compensators is developed. Also, the program is able to present a simulation of the UPS model in the graphical interface. Finally, practical tests results are presented, using the parameters obtained from the program in an actual UPS.

Key-words: *GUI* - Graphical user interface; Resonant controllers; UPS – Uninterruptible Power Supplies

## LISTA DE ILUSTRAÇÕES

Figura 1 - Típica carga não-linear: ponte completa de diodos .....	14
Figura 2 - Forma da tensão de saída da <i>UPS</i> (1), corrente em carga linear ligada à <i>UPS</i> (2) e corrente de retificador a ponte completa ligado à <i>UPS</i> (3) .....	14
Figura 3 - Esquema simplificado de uma <i>UPS</i> .....	18
Figura 4 - Esquema simplificado da etapa CC-CA .....	18
Figura 5 - Diagrama de blocos do conversor CC-CA .....	20
Figura 6 - Regiões <i>LMI</i> .....	21
Figura 7 - Janela inicial do <i>GUI-Guide</i> .....	22
Figura 8 - Blank <i>GUI</i> .....	23
Figura 9 - Blocos mais utilizados.....	25
Figura 10 - Blocos “ <i>Axes</i> ” e “ <i>Table</i> ” .....	27
Figura 11 - Interface gráfica principal.....	34
Figura 12 - Interface de abertura.....	35
Figura 13 - Carregar dados iniciais da <i>UPS</i> na interface principal .....	37
Figura 14 - Fluxograma do programa da <i>GUI</i> .....	41
Figura 15 - Sistema completo do controlador ressonante.....	42
Figura 16 - Interface <i>help1</i> .....	44
Figura 17 - Interface <i>help2</i> .....	44
Figura 18 - Interface gráfica “FECHAR” .....	45
Figura 19 - Fluxograma da <i>GUI</i> “ <i>Gui_Graf</i> ” .....	46
Figura 20 - Interface gráfica “ <i>Gui_Graf</i> ” .....	47
Figura 21 - Modelo do conversor CC-CA da <i>UPS</i> , no <i>PSIM</i> .....	49
Figura 22 - Modelo do sistema no Simulink, com bloco de ligação (em laranja) para comunicação com o <i>PSIM</i> .....	49
Figura 23 - Primeiro passo para obter ganhos do controlador.....	51
Figura 24 - Parâmetros de entrada, já aplicados à <i>GUI</i> principal.....	52
Figura 25 - Indicação de <i>pushbutton</i> para executar cálculo dos ganhos.....	52
Figura 26 - Resultados para a <i>UPS</i> de interesse, com um compensador ressonante .....	53
Figura 27 - Solicitar a execução da simulação .....	54
Figura 28 - Gráficos da tensão de saída e da tensão RMS de saída na <i>GUI</i> .....	54
Figura 29 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 0% de carga linear, um compensador ressonante .....	56

Figura 30 - THD da saída da <i>UPS</i> , 0% de carga linear, um compensador ressonante .....	56
Figura 31 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 80% de carga linear, um compensador ressonante.....	57
Figura 32 - THD da saída da <i>UPS</i> , 80% de carga linear, um compensador ressonante.....	57
Figura 33 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 75% de carga não-linear, um compensador ressonante .....	58
Figura 34 - THD da saída da <i>UPS</i> , 75% de carga não-linear, um compensador ressonante .....	59
Figura 35 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 0% de carga linear, dois compensadores ressonantes .....	60
Figura 36 - THD da saída da <i>UPS</i> , 0% de carga linear, dois compensadores ressonantes.	60
Figura 37 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 80% de carga linear, dois compensadores ressonantes .....	61
Figura 38 - THD da saída da <i>UPS</i> , 80% de carga linear, dois compensadores Ressonantes .....	61
Figura 39 - Forma de onda da tensão e corrente de saída da <i>UPS</i> , 75% de carga não-linear, dois compensadores ressonantes .....	62
Figura 40 - THD da saída da <i>UPS</i> , 75% de carga não-linear, dois compensadores ressonantes .....	62

## LISTA DE TABELAS

Tabela 1 – Parâmetros da <i>UPS</i> para testes .....	50
Tabela 2 – Parâmetros <i>LMI</i> utilizados para projeto do controlador .....	51
Tabela 3 – Ganhos do controlador com um compensador ressonante .....	53
Tabela 4 – Ganhos do controlador com dois compensadores ressonantes .....	55
Tabela 5 – Resultados para simulação/ensaio com carga linear .....	58

## LISTA DE ABREVIATURAS

CA-CC – conversor de corrente alternada para corrente contínua.

CC-CA – conversor de corrente contínua para corrente alternada;

*GUI* – “*graphical user interface*”, termo em inglês para “interface gráfica de usuário”;

*GUI-Guide* – Assistente de programação de Interface gráfica de usuário, no MatLab

*LMI* – “*linear matrix inequalities*”, termo em inglês para “desigualdades matriciais lineares”;

PID – “Proporcional, integral e derivativo”;;

*RMS* – “*Root Mean Square*”, termo em inglês para “valor eficaz”;

*THD* – “*total harmonic distortion*”, termo em inglês para “distorção harmônica total”;

*UPS* – “*Uninterruptible Power Supply*”, termo em inglês para “suprimento ininterrupto de potência” – chamado no Brasil de “*Nobreak*”;

## SUMÁRIO

1.	<b>INTRUDUÇÃO</b> .....	13
2.	<b>FUNDAMENTOS</b> .....	17
2.1.	<i>A UPS</i> .....	17
2.2.	CONTROLADORES RESSONANTES .....	19
2.3.	FUNDAMENTOS DA CONSTRUÇÃO DAS <i>LMI</i> .....	21
2.4.	COMPONENTES BÁSICOS DE UMA <i>GUI</i> .....	21
3.	<b>INTERFACE PARA CÁLCULO DE GANHOS</b> .....	29
3.1.	DADOS DE INTERESSE .....	30
3.1.1.	<b>Dados da <i>UPS</i></b> .....	30
3.1.2.	<b>Perfil de carga</b> .....	30
3.1.3.	<b>Dados dos controladores ressonantes</b> .....	31
3.1.4.	<b>Dados acerca da construção das <i>LMI</i></b> .....	32
3.1.5.	<b>Dados de saída da <i>GUI</i></b> .....	32
3.2.	INTERFACE GRÁFICA PROPOSTA .....	33
3.3.	CONJUNTO DE <i>GUI</i> .....	34
3.3.1.	<b>Interface de abertura</b> .....	35
3.3.2.	<b>Interface para cálculo dos ganhos</b> .....	36
3.3.3.	<b>Interface de ajuda para saída de dados</b> .....	43
3.3.4.	<b>Interface de ajuda para entrada de dados</b> .....	43
3.3.5.	<b>Interface com opção de simulação</b> .....	45
3.3.6.	<b>Interface com tensão de saída da simulação</b> .....	45
4.	<b>RESULTADOS</b> .....	48

4.1.	MÉTODO DE SIMULAÇÃO .....	48
4.2.	CÁLCULO DOS GANHOS PARA <i>UPS</i> .....	50
4.3.	IMPLEMENTAÇÃO DO CONTROLADOR E RESULTADOS .....	55
5.	<b>CONCLUSÃO</b> .....	64
5.1.	TRABALHOS FUTUROS .....	65
	<b>REFERÊNCIAS</b> .....	67
	<b>ANEXO – PRINCIPAIS “LINHAS DE PROGRAMAÇÃO” NO <i>GUI-GUIDE</i></b> .....	70

## 1. INTRODUÇÃO

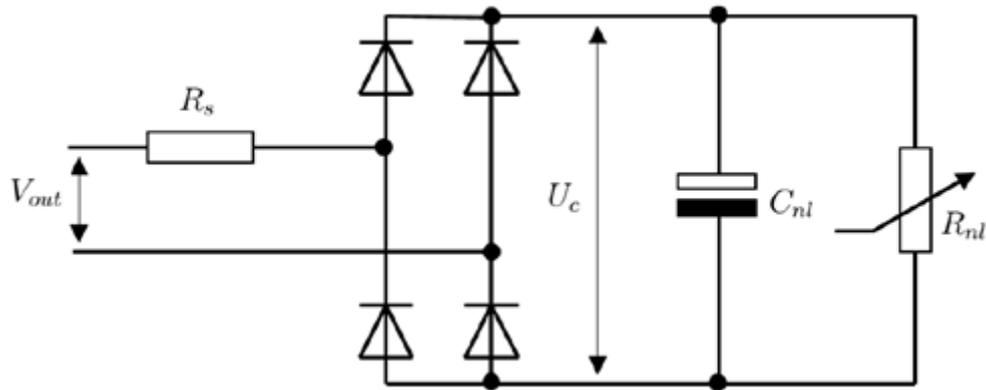
Fontes ininterruptas de energia (*UPS*, *Uninterruptible Power Supply*) são equipamentos que devem ser capazes de manter em sua tensão de saída as mesmas características da tensão disponibilizada pela rede elétrica a qual está conectada: frequência constante, amplitude de tensão constante e forma de onda senoidal – mesmo quando a rede primária apresentar qualquer falha (distorção, ou mesmo corte no suprimento - *blackout*). Na prática, busca-se que, em caso de falta de energia elétrica, por exemplo, os equipamentos conectados à saída da *UPS* continuem operando normalmente. O método básico utilizado para se obter os resultados propostos é comparar a tensão de saída da *UPS* com um sinal de referência senoidal ideal – um sinal com forma de onda igual à forma de onda esperada da rede elétrica primária em funcionamento normal.

Entretanto, o seguimento do sinal de referência se torna complexo devido à natureza não linear associada a diversos tipos de carga. Por exemplo, a grande maioria dos dispositivos eletrônicos requerem fontes de corrente contínua, dotados de um estágio retificador com diodos em ponte completa – uma grande fonte de distorções harmônicas devida à forma de onda da corrente e, conseqüentemente, da tensão nos cabos da rede.

É apresentada na Figura 1 [1] a imagem de uma ponte de diodos, retificador muito comum na entrada de diversos dispositivos eletrônicos. Na Figura 2, é apresentada a forma de onda senoidal, bem como a forma de onda da corrente em uma carga linear e a forma de onda da corrente quando a tensão senoidal é aplicada ao retificador da Figura 1.

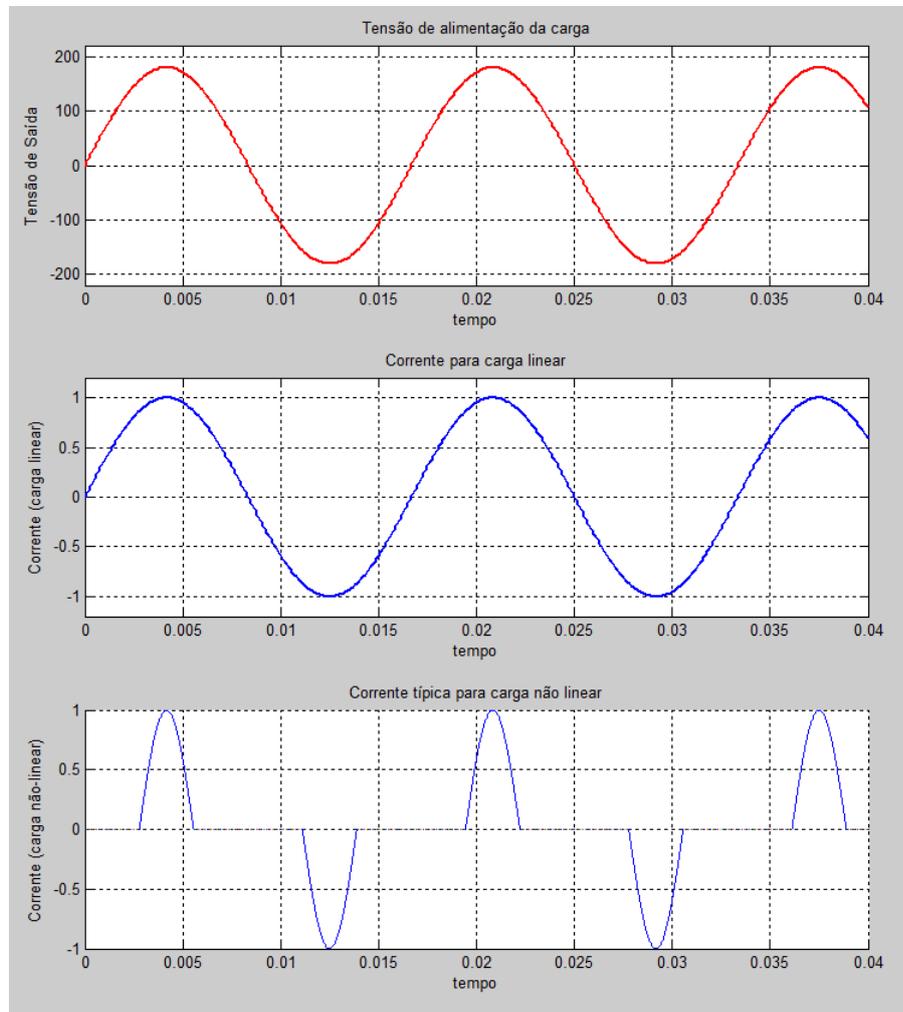
Diferentes controladores podem ser aplicados no sistema da *UPS*, incluindo os controladores PID (Proporcional Integral Derivativo), extremamente difundidos na indústria. Estes, entretanto, baseiam seu algoritmo de controle no valor da tensão RMS (*Root Mean Square*) para obtenção do sinal de realimentação. Assim, a utilização de controladores PID resolve o problema da amplitude de tensão, mas, de forma geral, não resolve o problema da distorção da forma de onda da tensão de saída.

Figura 1. Típica carga não linear: ponte completa de diodos



Fonte: IEC, 2011

Figura 2. Forma da tensão de saída da UPS (1), corrente em carga linear ligada à UPS (2) e corrente de retificador da ponte completa ligado à UPS (3).



Fonte: Gerado pelo autor no software MatLab

Uma alternativa para resolver a distorção harmônica por cargas não lineares (ou adição/remoção abrupta de carga) é o emprego de compensadores ressonantes, ajustados para

atuar na frequência fundamental de interesse e as harmônicas mais significativas. Este método é baseado na adição de modos harmônicos (funções seno ou cosseno) na malha direta de controle. O controlador ressonante ideal apresenta ganho infinito na frequência de ressonância, rejeitando as demais componentes. O uso de determinado número de controladores ressonantes em paralelo permite a rejeição de perturbações ou seguimento de referência nas determinadas frequências.

Entretanto, um elevado número de compensadores ressonantes acaba por criar um sistema cuja sintonia se torna complexa. Ainda, deve-se considerar as faixas de carga e variação de carga que se tem interesse na formulação do problema.

Para resolver este problema, conforme [2], foi proposto um método baseado na solução de um problema de otimização com restrições *LMI* (*linear matrix inequalities*), que permite calcular de forma concatenada os parâmetros de diferentes compensadores ressonantes atuando em frequências diferentes. O procedimento proposto inclui a solução para uma faixa de carga (e não para apenas uma carga com potência preestabelecida), e torna-se adequado para aplicação nas *UPS*.

Busca-se um método aplicável a qualquer *UPS*, desde que sejam conhecidos os parâmetros básicos da mesma. Consequentemente, torna-se útil uma interface que realize de forma rápida a solução do problema proposto e apresente ao usuário os valores dos ganhos dos compensadores ressonantes. A proposta deste trabalho é criar uma interface que apresente os ganhos do controlador ressonante para uma *UPS* conhecida, para determinada capacidade de faixa de carga aplicada.

A solução do problema de otimização com restrições *LMI* será realizada conforme proposta em [3], por meio de código implementado no MatLab® (software matemático, por MathWorks). A interface deverá apresentar em um único ambiente, de fácil compreensão, todos os dados que o usuário deverá conhecer sobre a *UPS* em questão, assim como a faixa de carga que deseja aplicar na mesma. Esses dados devem ser utilizados no código para cálculo dos ganhos e os resultados devem ser apresentados no ambiente gráfico.

Os compensadores ressonantes têm a forma apresentada na Equação (1), e a quantidade de componentes harmônicas que se deseja controlar determina a quantidade de compensadores ressonantes necessários. Os valores de A e B são os ganhos do controlador, e  $w$  é a frequência de ressonância do filtro.

$$G(s) = \frac{As+B}{s^2+w^2} \quad (1)$$

Dessa forma, deseja-se que a interface gráfica apresente os valores de A e B adequados para cumprir os requisitos de desempenho em cada frequência  $w$  de aplicação dos compensadores harmônicos.

Para implementação da interface gráfica, se utilizará a aplicação “*GUI Guide*” (*Graphical User Interface Guide*), disponibilizada no MatLab®. Esta aplicação permite a edição da parte visual da *GUI* de forma simplificada. Diversos blocos são apresentados ao usuário (os blocos que serão utilizados serão abordados durante o desenvolvimento), e um segmento de código é automaticamente criado. O ambiente de trabalho do MatLab® ou o ambiente de trabalho interno da *GUI* devem ser completamente transparentes ao usuário.

O modelo básico da *UPS* é apresentado e discutido, a fim de apontar os parâmetros de interesse, bem como os blocos disponibilizados pelo *GUI Guide* que serão utilizados na interface proposta. Em um segundo momento, apresenta-se o fluxograma da *GUI*, juntamente com os resultados deste trabalho. Finalmente, apresenta-se um exemplo com os dados de uma *UPS* real sendo controlada por compensadores ressonantes com ganhos calculados a partir da interface gráfica.

## 2. FUNDAMENTOS

No capítulo 1, definiu-se que o trabalho aqui proposto é uma forma de obter os ganhos de um controlador ressonante para uma *UPS*, através de uma interface gráfica. Aqui, os termos importantes ao desenvolvimento deste trabalho são apresentados e discutidos. Detalhes sobre que é uma *UPS*, o que é o controlador ressonante, bem como quais são os critérios utilizados na construção das restrições *LMI*, são apresentados.

### 2.1. *UPS – UNINTERRUPTIBLE POWER SUPPLY*

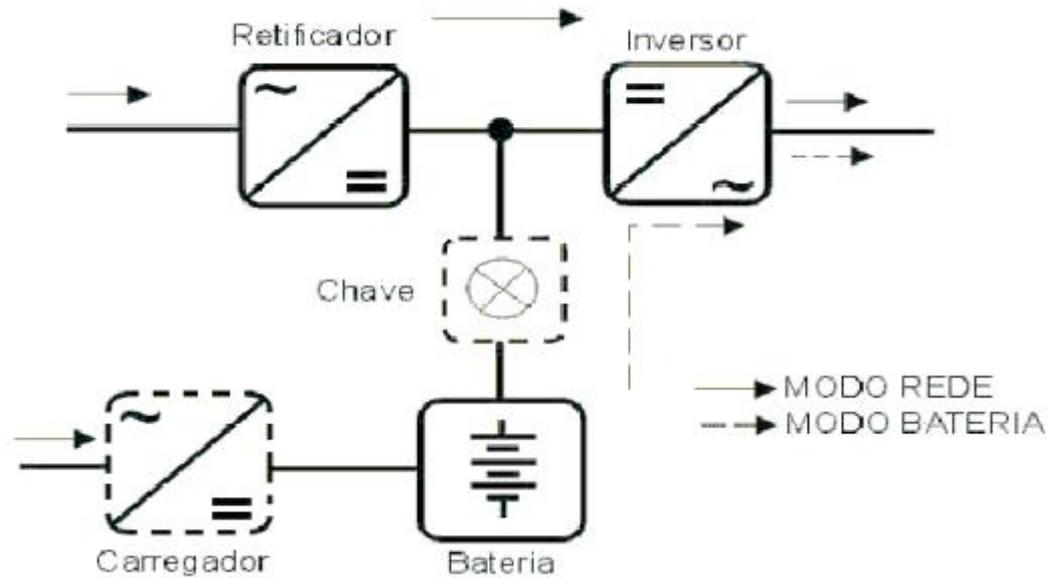
Equipamento normalmente denominado Nobreak no Brasil, a *UPS* é um dispositivo que mantém a tensão de saída dentro de limites satisfatórios para determinada faixa de carga. Idealmente, respeitando-se a potência para o qual a *UPS* é projetada, a forma de onda da tensão não irá variar. A forma de onda da tensão de saída senoidal é uma situação ideal, e significaria uma THD (distorção harmônica total) nula, bem como nenhuma variação na amplitude da onda de saída.

Fisicamente, os blocos integrantes de uma *UPS* de dupla conversão (também chamados de “*online*”, por suprirem a alimentação da carga mesmo durante falhas na rede primária) podem ser resumidamente descritos como um conversor CA-CC ligando a rede elétrica primária ao banco de baterias, que alimenta um segundo conversor CC-CA que fornece a tensão de saída da *UPS*. Um esquema simplificado da *UPS* em questão nesse trabalho pode ser observado na Figura 3, conforme [4].

O conversor CC-CA é a etapa cujo algoritmo de controle é descrito neste trabalho, e onde encontra-se a maior dificuldade do projeto. Na prática, devido às características não lineares das cargas, é impossível obter o comportamento ideal. Ainda, a adição (ou remoção) de carga causa distorções harmônicas devido à variação transitória, mesmo que tal carga seja puramente resistiva (linear). Assim, o que se busca na prática é tornar o sistema robusto, capaz de suportar variação de carga e suprir cargas não lineares cumprindo requisitos de máxima distorção na forma de onda da tensão de saída.

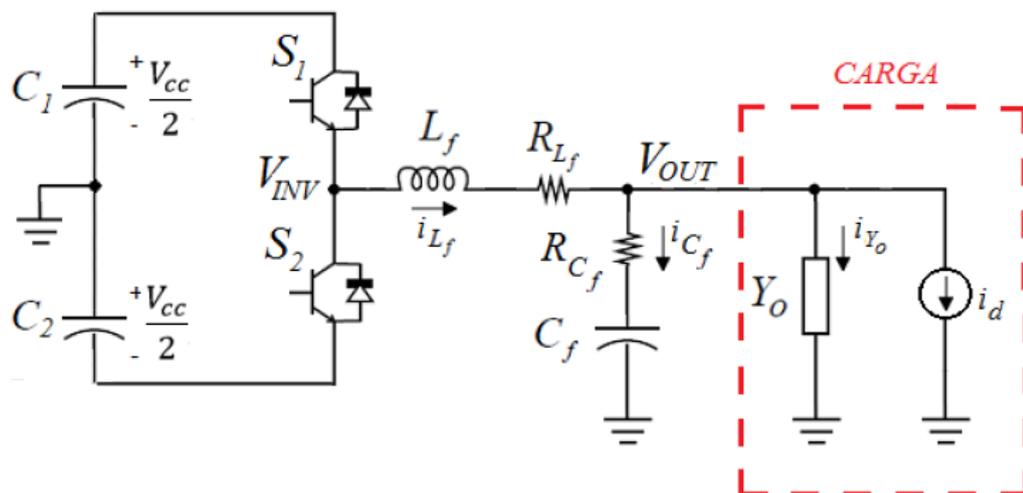
Um esquema simplificado da etapa CC-CA é apresentado na Figura 4. Nela, os componentes indutivos, capacitivos e resistivos relacionados à *UPS* estão explícitos para um conversor seguindo a topologia meia ponte.

Figura 3. Esquema simplificado de uma UPS



Fonte: ABNT, 2013

Figura 4. Esquema simplificado da etapa CC-CA



Fonte: [5]

Na Figura 4, temos na saída do conversor CC-CA três componentes do filtro de saída da UPS, a admitância  $Y_o$ , representando a carga linear e  $I_d$ , representando o dreno de corrente relacionado à carga não-linear. É válido destacar que existem diversas topologias para a etapa CC-CA de uma UPS, com modelos matemáticos distintos. Tais UPS não poderiam ser

controladas adequadamente pelos compensadores ressonantes cujos ganhos podem ser calculados com o uso da interface gráfica aqui proposta.

## 2.2. CONTROLADORES RESSONANTES

Por ser altamente difundido e conhecido, com estrutura simples e métodos desenvolvidos para seu projeto, o controlador Proporcional-Integral-Derivativo, PID, ainda é o mais utilizado em *UPS* comerciais [6], [7]. Porém, este controlador é adequado para seguimento de referências constantes, tipo zero, não apresentando bons resultados para seguimento de sinais de referência senoidais. Para possibilitar o uso deste controlador em sistemas como a *UPS*, geralmente utiliza-se o sinal de tensão RMS na malha de realimentação. Desta forma, este controlador não busca compensar a distorção harmônica gerada por cargas não lineares.

Com o objetivo de desenvolver métodos de controle eficazes em assegurar o seguimento de uma referência senoidal, bem como rejeição de perturbações relacionadas a cargas não lineares, novas metodologias vêm sendo desenvolvidas. Aqui, os controladores ressonantes [8], baseados no Princípio do Modelo Interno [9], são utilizados.

Os compensadores ressonantes ideais apresentam ganho infinito na frequência de ressonância. O uso de múltiplos compensadores ressonantes permite a alocação de ganho em múltiplas frequências de interesse. De forma simplificada, pode-se aplicar um compensador ressonante para seguimento da frequência fundamental. Para rejeição de cada frequência múltipla harmônica da fundamental, é possível aplicar novos compensadores ressonantes.

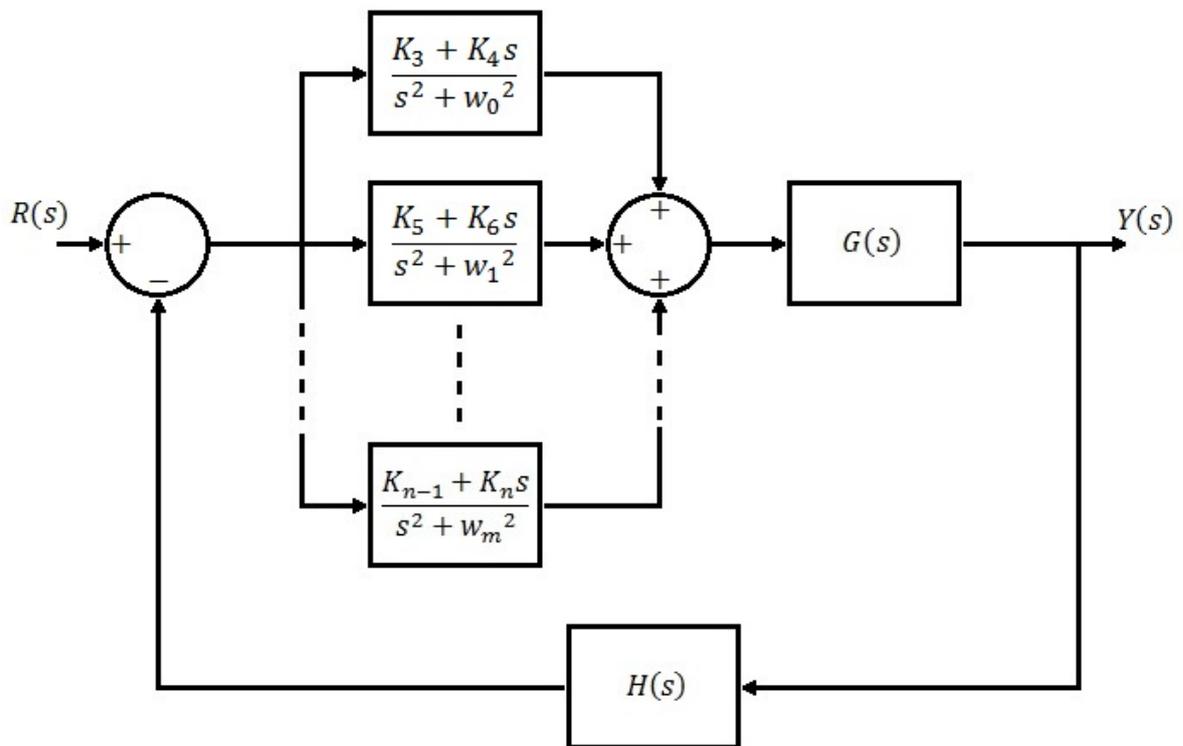
Como cargas reais não drenam corrente de forma linear (com forma de onda igual à forma de onda da tensão aplicada), a utilização de apenas um compensador na frequência fundamental não é suficiente para manter a THD satisfatoriamente baixa. Conseqüentemente, utiliza-se um conjunto de compensadores ressonantes.

As cargas “referência” para este trabalho são apresentadas na Figura 2. Para estas cargas de interesse, utiliza-se além de um compensador ressonante para a frequência fundamental, compensadores ressonantes para as componentes de maior contribuição harmônica. Em geral, são os múltiplos ímpares de menor frequência: 3ª harmônica, 5ª harmônica, 7ª harmônica, 9ª harmônica e 11ª harmônica. Para a corrente de saída não linear

apresentada na Figura 2, as componentes harmônicas de maior amplitude acontecem na 3ª, 5ª e 9ª harmônicas.

Para representar os múltiplos compensadores, a Equação (1) que modela de forma geral o compensador ressonante é repetida em paralelo no diagrama de blocos do sistema de controle, conforme apresentado na Figura 5. O diagrama de blocos do sistema do inversor é apresentado na figura, destacando os compensadores ressonantes na figura,

Figura 5. Diagrama de blocos do conversor CC-CA



Fonte: adaptado de [10]

O diagrama de blocos do sistema do inversor é sintetizado na Figura 5, sendo  $R(s)$  uma função de referência senoidal,  $Y(s)$  a tensão de saída da UPS,  $G(s)$  a planta do inversor,  $H(s)$  o ganho ou função aplicado na malha de realimentação, 'n' o índice dos ganhos dos compensadores ressonantes e 'm' o número de frequências harmônicas compensadas. Ainda, repara-se que os índices do primeiro compensador ressonante são 'K3' e 'K4'. Esse dado é assim representado devido a existência de dois valores relacionados ao erro do sistema e realimentação de corrente, que recebem os índices 'K1' e 'K2', conforme será explicado em capítulo posterior.

### 2.3. FUNDAMENTOS DA CONSTRUÇÃO DAS *LMI*

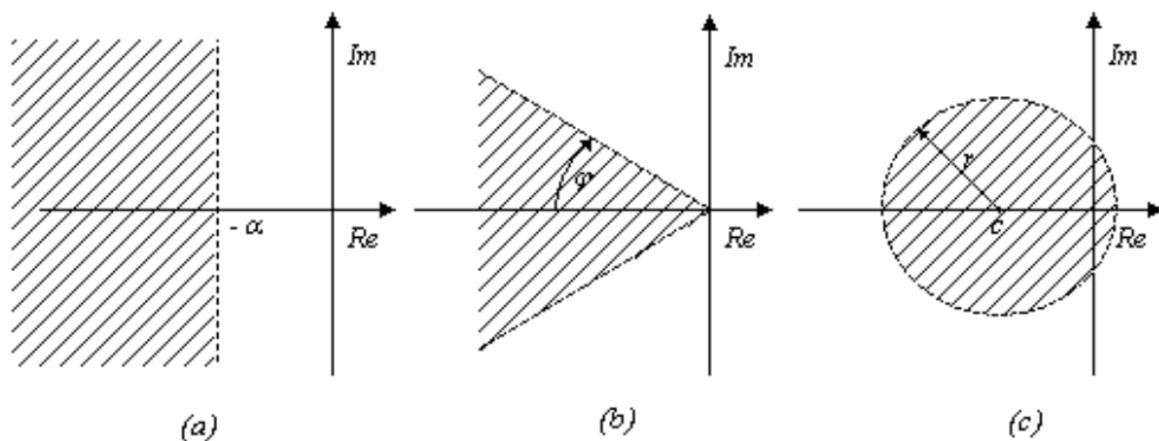
Conforme será abordado em seções futuras, o usuário da interface gráfica para cálculo dos ganhos dos controladores terá como opção a edição de alguns dos parâmetros para as restrições *LMI*. Conforme [2] e [10], a sintonia dos compensadores ressonantes deverá atender aos critérios de D-Estabilidade.

Tais critérios determinam três regiões no plano complexo:

- Região  $D_a$ , com parte real menor que  $-\alpha$ ;
- Região  $D_b$ , setor cônico definido pelo ângulo  $\varphi$ ;
- Região  $D_c$ , disco definido pela posição do centro e raio.

Os três critérios podem ser representados, de forma independente, pelas zonas com hachuras na Figura 6. Define-se que a intersecção das três zonas define a região com D-Estabilidade.

Figura 6. Regiões *LMI*



Fonte:[2]

### 2.4. COMPONENTES BÁSICOS DE UMA *GUI*

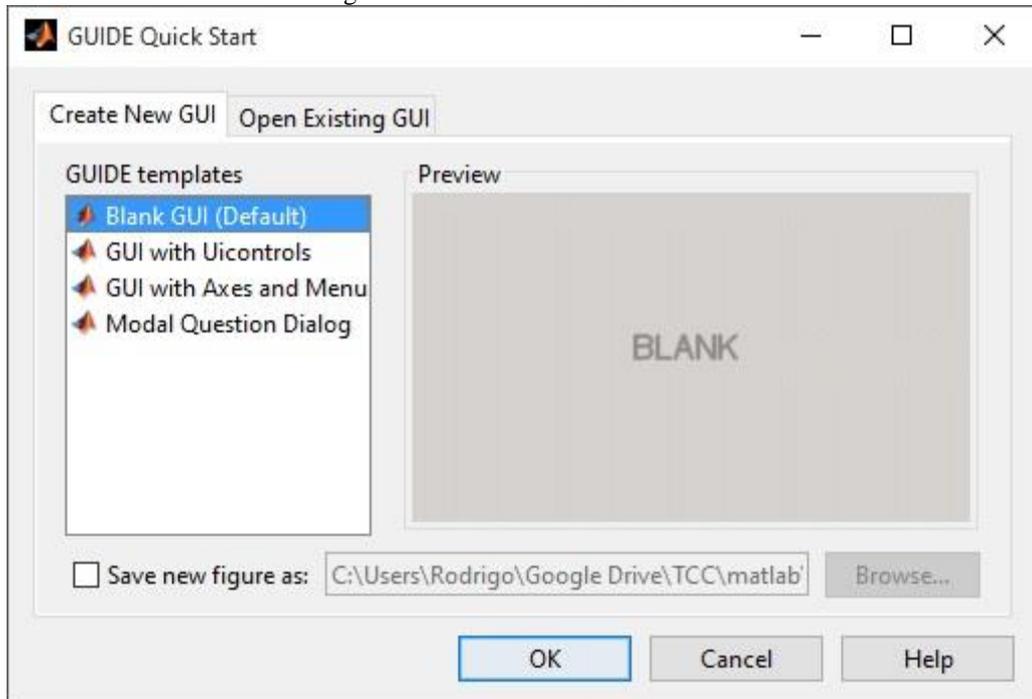
Aqui, como se pretende propor uma *GUI* no ambiente MatLab®, os dados e instruções são somente válidos para esse caso. Outras metodologias e abordagens, podendo incluir programação em diferentes linguagens, poderiam ser utilizadas para solucionar o mesmo

problema. Justifica-se o uso do MatLab pois para o mesmo, já existem *scripts* que calculam eficientemente os ganhos do controlador ressonante.

Conforme [11] no console do MatLab®, se digita o comando “*guide*”, para abrir a aplicação “*GUI-Guide*”. A aplicação *GUI Guide* é o editor de *GUI* disponível no MatLab®, e com o mesmo, se pode editar/criar a parte gráfica de uma *GUI* de forma simplificada. Executando a edição/criação de uma *GUI* com o *GUI Guide*, o MatLab® automaticamente cria o cabeçalho do código que contém cada bloco utilizado, sem nenhuma função associada (o programador irá definir as funções de cada bloco).

Após executar o comando “*guide*”, a janela inicial da aplicação se torna visível. A mesma é apresentada na Figura 7.

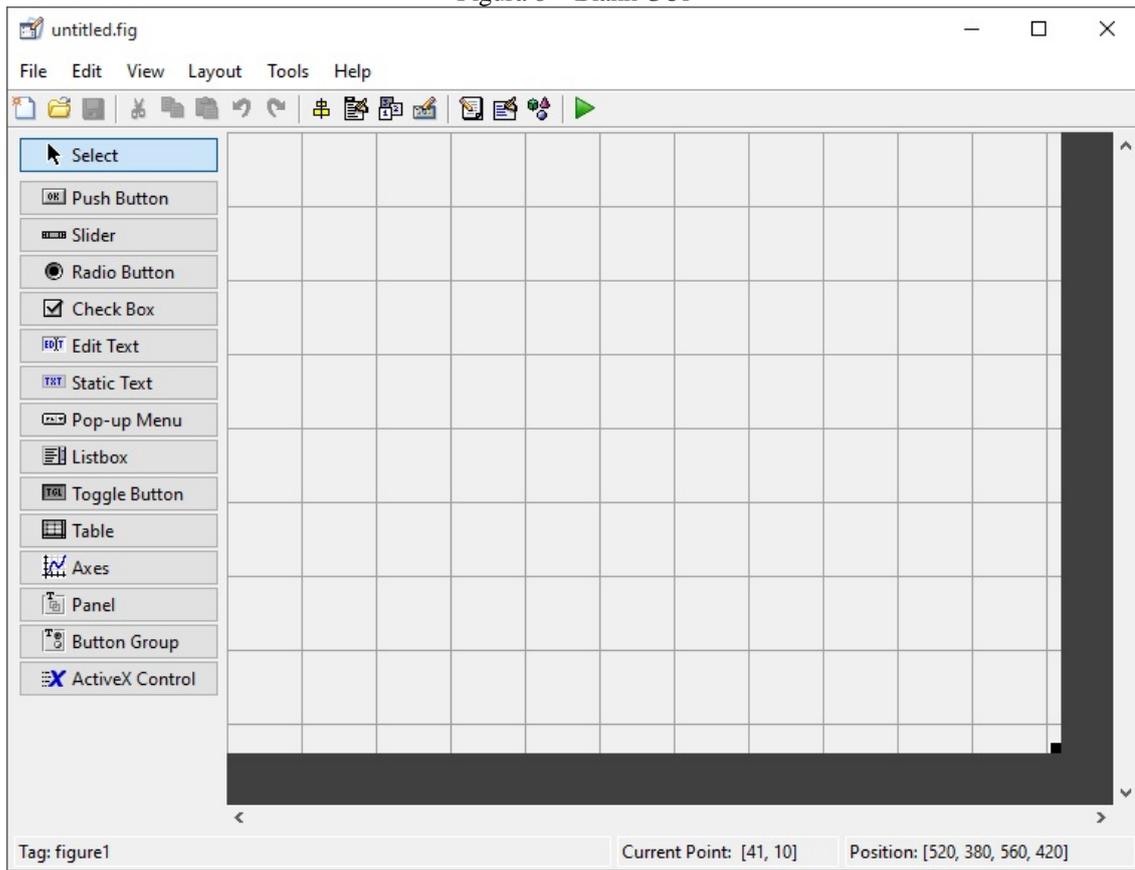
Figura 7. Janela inicial do *GUI-Guide*



Fonte: Gerado pelo autor, através do software MatLab

Seleciona-se a primeira opção, “*Blank GUI (Default)*”, e se clica em “*OK*”. Assim, se abre uma *GUI* vazia, com uma lista de blocos à esquerda. A janela inicial da “*Blank GUI*” é apresentada na Figura 8. Uma vez salva a *GUI* criada, a mesma irá automaticamente criar uma função do MatLab®.

Figura 8 – Blank GUI



Fonte: Gerado pelo autor, através do software MatLab

Cada bloco do *GUI* criado tem uma grande quantidade de parâmetros, que podem ou não ser pertinentes no programa de interesse. O parâmetro mais utilizado na lógica do sistema será o “TAG”, parâmetro contendo o nome pelo qual o bloco atende dentro do código. Por exemplo, quando deseja-se escrever determinado dado em um dos blocos, o envio desse dado é feito destinando o mesmo para o “TAG” associado ao bloco.

Todos os blocos apresentados neste subcapítulo foram retirados de [11]. São apenas apresentados exemplos básicos, como forma de introduzir as possibilidades do recurso de construção de *GUI* através do MatLab®. Em um momento posterior, apresenta-se o fluxograma dos segmentos de maior interesse do código.

Clica-se no bloco desejado, à esquerda da *Blank GUI*, com o botão esquerdo do mouse. Em seguida, clica-se em um ponto qualquer dentro da *Blank GUI*, e nessa posição, será adicionado o bloco desejado.

Na Figura 9, observam-se alguns dos blocos mais importantes aplicadas sobre a “*Blank GUI*”. Os demais blocos que serão explicadas subsequentemente estão apresentados sobre a “*Blank GUI*” na Figura 10.

## CALLBACK

Clicando-se com o botão direito do mouse sobre qualquer bloco que tenha sido adicionado à *blank-GUI*, algumas opções aparecem ao seu lado. Uma delas é a guia “*Call-back*”, que se selecionada, apresentará novas opções.

O *callback* é um código próprio de cada bloco, que pode ser executado de diferentes formas, dependendo do tipo de bloco e da vontade do programador. Ainda que existam muitos tipos de *call-back*, aqui só serão apresentados os tipos que serão utilizados:

- *Call-back*: no contexto desse trabalho, o *call-back* do tipo *call-back* é o mais importante. Ele irá executar uma sequência de código quando o usuário clicar sobre o bloco o qual se tenha designado uma função.

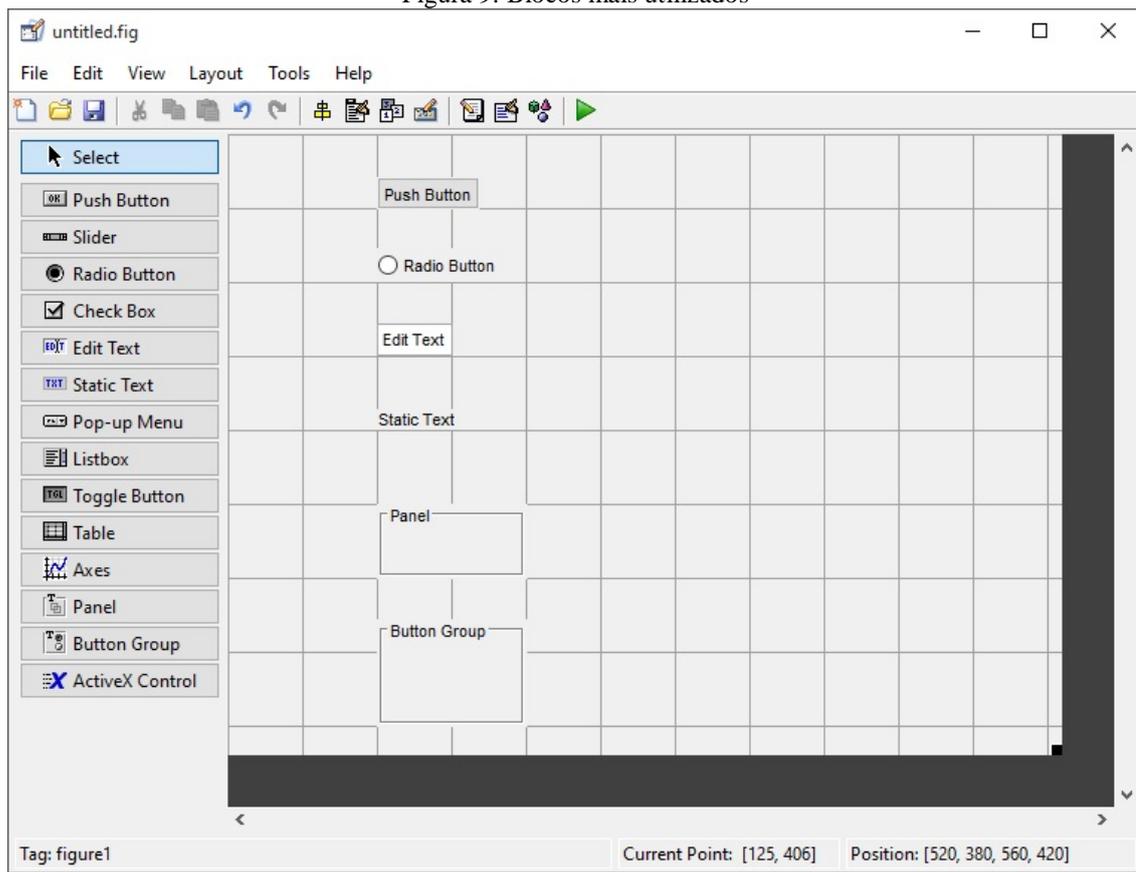
- *CellCelectionCall-back*: irá executar segmento de código quando uma célula de uma tabela for selecionada.

Blocos passivos, como o caso da “*Static Text*”, não possuem um “*Call-back*” que execute uma função – ou seja, mesmo clicando sobre tal bloco, o mesmo não executa nenhum segmento de código de programação. Ainda, para os blocos que podem executar alguma sequência de código quando selecionadas, não é obrigatório que o programador designe qualquer lógica para as mesmas. Em ANEXO, são apresentadas as principais estruturas de código que podem ser utilizadas dentro do código de uma *GUI* do MatLab®.

## EDIT TEXT - CAIXA DE ENTRADA E SAÍDA DE TEXTO

A caixa de entrada e saída de texto, “*edit text*” na lista de funções, permite que se transfira para o ambiente de trabalho da *GUI* (a *GUI* possui um “ambiente” próprio, onde ficam as variáveis utilizadas pela mesma – separado do ambiente das variáveis do console do MatLab®) um dado no formato “*string*” [14]. A mesma possui *Call-back*, que irá executar uma determinada sequência de código quando um dado for introduzido na caixa, ou quando a mesma for selecionada.

Figura 9: Blocos mais utilizados



Fonte: Gerado pelo autor, através do software MatLab

## STATIC TEXT - CAIXA DE TEXTO ESTÁTICO

A caixa de texto estático é um indicador, e deverá ser usada basicamente para informar o usuário acerca de funções/situações dentro da interface. Como o nome sugere, esse tipo de bloco não recebe dados do usuário, apenas exibe uma informação, no formato ‘*String*’.

Contudo, uma caixa de texto estático pode ser editada pelo programa da *GUI*. Ainda assim, neste caso, o texto estático estaria apenas transmitindo informações para a interface – nunca lendo dados que o usuário inseriu, mantendo a natureza passiva. Este bloco não apresenta *Call-back* de função. Isso acontece porque este bloco não pretende executar sequência alguma de código.

## RADIO-BUTTON

O bloco “*radio-button*” é um botão que pode ser selecionado ou não, e sua informação pode ser captada em lógica booleana com níveis ‘zero’ ou ‘um’. Este bloco possui *Call-back* que permite a execução de segmento de código quando a mesma for selecionada.

## PANEL – PAINEL DE BLOCOS

O painel de blocos, apresentado na Figura 9, permite organizar a interface gráfica, separando grupos de blocos de acordo com a vontade do programador. Por exemplo, pode-se colocar varias caixas de texto “*Edit text*” dentro de um painel, e definir que todos esses blocos serão relacionadas às entradas de dados da *GUI*.

O painel não possui a possibilidade de executar uma função através de *Call-back*, visto que ele é um bloco que, sobretudo, organiza a *GUI*. Porém, os blocos internos podem executar função através de *Call-back* (se possuírem essa funcionalidade).

## BUTTON GROUP - GRUPO DE BOTOES MUTUAMENTE EXCLUSIVOS

O grupo de botões mutuamente exclusivos é uma área fechada, como se observa na Figura 9. Dentro dessa área delimitada por uma linha escura, podem ser adicionados mais de um “*radio-button*”. Eles irão adquirir a propriedade de ser mutuamente exclusivos. Nesse caso, quando se selecionar um dos “*Radio-Button*”, este será colocado em ‘um’ (ligado), e todos os demais serão colocados em ‘zero’ (desligado).

É possível criar um *Call-back* para este bloco. No caso, um segmento de programação será executado sempre que se selecionar uma das caixas dentro do grupo de botões.

## PUSH BUTTON - BOTÃO

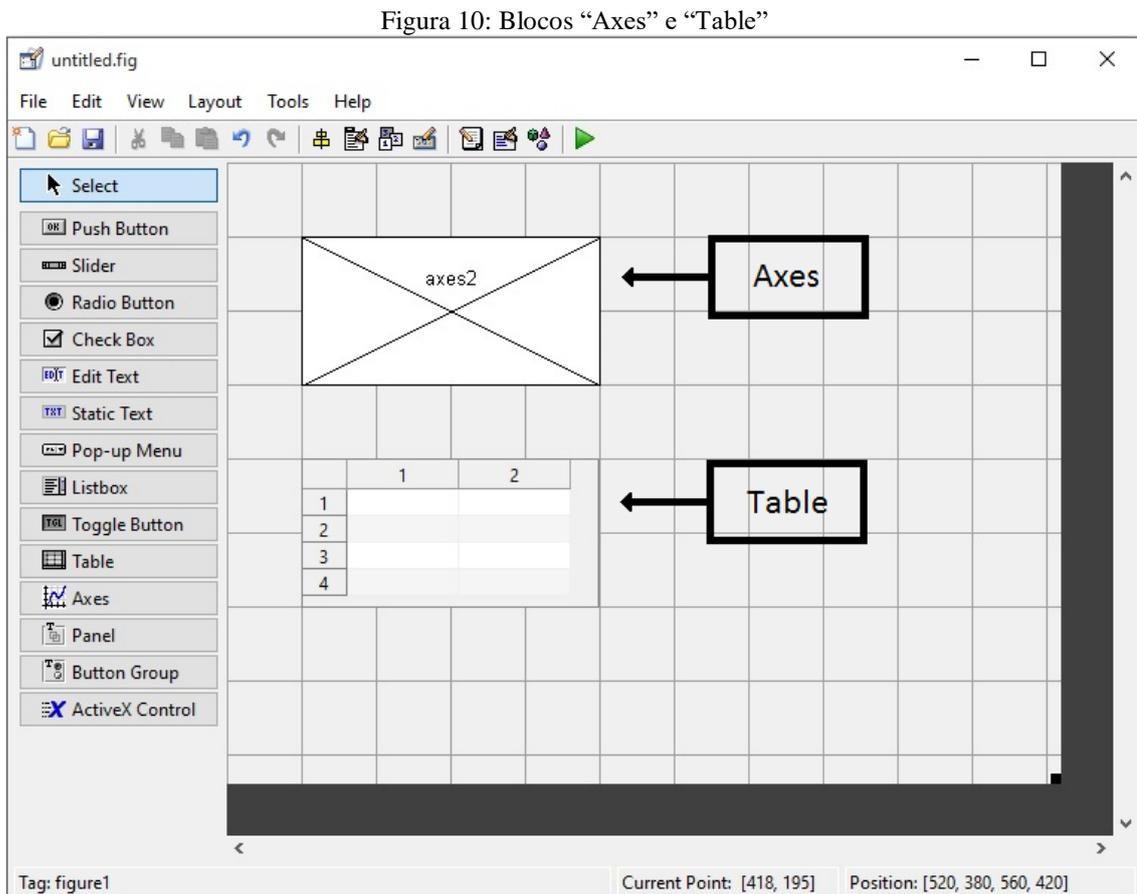
O *Push Button* (Figura 9) basicamente executa uma função. Ainda que normalmente seja possível fazer com que o código da *GUI* rode utilizando os *Call-back's* de outros blocos, tal abordagem seria inadequada. A maioria dos callbacks de outros blocos executam um código após alguma edição qualquer. Depois de cada edição de dados dentro de cada “*edit text*” por exemplo, o código rodaria por completo. Se existem várias funções “*edit text*” dentro da *GUI*, editar cada uma delas acarretaria na execução de um segmento de código quando normalmente, se tem interesse em realizar a execução apenas após todas as edições.

De forma geral, não se utiliza a possibilidade de executar funções complexas pelas *call-backs* de outros blocos senão a do *Push Button*, pois dessa forma, o usuário tem a decisão de quando todas as edições nos blocos de entrada de dados foram realizadas, momento em que o código deve ser de fato executado.

## TABLE - TABELA

Ainda que seja possível exibir em “*static text*” até mesmo vetores, com vários dados simultaneamente, essa é uma alternativa desorganizada se forem muitos os dados de interesse. Para tal, é possível utilizar uma tabela, como é apresentada na Figura 10.

Com a tabela, pode-se exibir uma matriz 4x4, por exemplo, de forma muito organizada e em um pequeno espaço dentro da *GUI*. Ainda, a transferência dos dados para o bloco “*Table*” é feita apenas transferindo uma variável de dimensões compatíveis para a mesma. No caso do uso de *Static Text*, seria necessário segmentos de programa específicos para cada um dos dados.



Fonte: Gerado pelo autor, através do software MatLab

## AXES - GRÁFICO

Apresentada sobre uma *Blank-GUI* na Figura 10, esse bloco permite colocar na *GUI* gráficos de qualquer tipo que possa ser plotado no MatLab®, e até mesmo figuras. É um bloco de saída de dados, e não apresenta possibilidade de executar uma função através de *Call-back*.

Este bloco será usada como qualquer figura em *Script* de MatLab®. Basta preparar a imagem e área de plotagem, e então utilizar o segmento adequado de código que reproduz na interface tal imagem.

### 3. INTERFACE PARA CÁLCULO DOS GANHOS

Como explicado, a proposta do presente trabalho é apresentar uma interface que permita o cálculo dos ganhos de um controlador do tipo ressonante, mantendo o MatLab® e qualquer tipo de código de programação transparente ao usuário, tornando fácil a obtenção dos parâmetros do controlador. Isso implica em uma *GUI* que tenha como parâmetros de entrada dados da *UPS* (incluindo algumas opções para o controlador), e como saída, os parâmetros do controlador.

Na prática, alguns dados acerca do controlador deverão ser decididos pelo usuário da interface. Como foi citado anteriormente, o usuário deve decidir quantos compensadores ressonantes serão utilizados. Ainda, parâmetros relacionados às restrições *LMI* poderão ser editados pelo usuário.

Os dados de saída de maior interesse são os valores constantes aplicados aos compensadores ressonantes. Tais valores correspondem aos valores de *A* e *B* na Equação 1, reproduzida abaixo:

$$G(s) = \frac{As + B}{s^2 + w^2}$$

O *script* proposto em [3] é capaz de solucionar o problema envolvendo as restrições *LMI*, fornecendo ao usuário os parâmetros do controlador ressonante. Dessa forma, na composição da interface, existem duas abordagens viáveis:

- 1) compor um único código para a *GUI*, que tenha um *call-back* contendo todo o código do *script*, ou;
- 2) compor uma *GUI* que tenha um *call-back* capaz de capturar os dados da interface e enviá-los para o *script*, e em seguida, recuperar a solução.

Para manter o código da *GUI* o mais eficiente possível, opta-se pela segunda abordagem. Dessa forma, propõe-se capturar os dados necessários através da interface gráfica, enviar os dados para o *script* de cálculo dos ganhos e apresentar na interface gráfica os resultados obtidos.

### 3.1. DADOS DE INTERESSE

Dados de interesse são ambos dados de entrada e saída. Logo, são "dados de interesse" toda a informação que deve ser conhecida e adicionada à *GUI*, bem como os parâmetros do controlador ressonante.

#### 3.1.1. Dados da *UPS*

Conforme discutido, os parâmetros da *UPS* são parte dos dados de entrada. Os dados pertinentes são os componentes do filtro de saída do conversor CC-CA da *UPS*, apresentados na Figura 4, como segue:

$C_f$  – Capacitância do filtro, em  $\mu\text{F}$  (microFarad);

$L_f$  – Indutância do filtro, em mH (miliHenry);

$R_{cf}$  – Resistência da capacitância do filtro, em  $\text{m}\Omega$  (miliOhm).

$R_{lf}$  – Resistência da indutância do filtro.

A rigor, há uma resistência associada à capacitância do filtro, mas essa costuma ser muito próxima de  $0\Omega$ . Ainda que esse parâmetro seja muito próximo de zero na grande maioria das *UPS*, o usuário poderá inserir seu valor.

Ainda, a potência, a tensão de saída RMS, bem como a frequência da tensão de saída, são parâmetros fundamentais da *UPS* e são parte dos parâmetros de entrada da *GUI*, conforme segue:

Potência – Potência nominal da *UPS*, em VA (Volt Ampére);

$V_{out}$  – Tensão RMS de saída da *UPS*, em V (volt);

Frequência – Frequência de saída, em Hz (hertz).

#### 3.1.2. Perfil de carga

Este trabalho versa, sobretudo, sobre as *UPS* que alimentam um retificador baseado em ponte de diodos, com grande índice de distorção harmônica na forma da tensão de saída

do conversor CC-CA. Ainda, é necessária resposta adequada para cargas lineares. Assim, o controlador projetado com a interface gráfica deverá ser capaz de atender adequadamente ambos os casos.

Os resultados obtidos com a interface aqui proposta serão testados em *UPS* de potência nominal de 3,5kVA. Ainda, o método de aplicação de carga durante os ensaios, como apresentado em capítulo posterior, é baseado em [12], este conforme [13] e [1], para a faixa de carga da *UPS* de interesse.

Para a potência citada, os ensaios devem apresentar saltos de 20% e 80% para carga linear, e 25% e 75% para carga não linear. Assim, para os dois tipos de carga, a interface gráfica irá prever a possibilidade de dois degraus de aplicação de carga, além do funcionamento da *UPS* à vazio.

Estes parâmetros serão apresentados como:

Carga linear:

1º degrau de carga – carga total aplicada antes da aplicação do segundo degrau de carga, em %;

2º degrau de carga – carga total aplicada à saída da *UPS* após uma segunda introdução de carga, em %.

Carga não-linear:

1º degrau de carga – carga total aplicada antes da aplicação do segundo degrau de carga, em %;

2º degrau de carga – carga total aplicada à saída da *UPS* após uma segunda introdução de carga, em %.

### **3.1.3. Dados dos controladores ressonantes**

Pode-se optar por manter o sistema mais simples, adicionando um controlador ressonante apenas na primeira harmônica da frequência fundamental. Este sistema, entretanto, não seria funcional se uma carga como a da Figura 1 fosse aplicada a saída – para esta carga,

por exemplo, sabe-se que existe distorção significativa associada até a nona harmônica da fundamental [4].

Assim, o objetivo é possibilitar a configuração para obtenção dos ganhos relacionados ao uso de múltiplos compensadores ressonantes em frequências ímpares da fundamental, até a décima primeira componente, através desta *GUI*. Ainda que o *script* seja capaz de calcular os ganhos do controlador ressonante para um número maior de componentes, para a carga não-linear de interesse nesse trabalho, compensadores ressonantes para frequências acima de 660Hz praticamente não surtiriam efeito algum.

Assim, o parâmetro relacionado aos compensadores ressonantes é:

Harmônicas – número de compensadores ressonantes aplicados à malha de controle,

#### **3.1.4. Dados acerca da construção das *LMI***

Uma breve discussão sobre os parâmetros da D-Estabilidade do sistema foi apresentada, e uma representação gráfica das zonas *LMI* dentro do plano complexo é mostrada na Figura 6. Alguns destes parâmetros são editáveis no código ‘roda\_teste’, e são disponibilizados para edição pelo usuário da interface gráfica:

-Alfa;

-Raio da circunferência;

-Centro da circunferência;

O parâmetro angular, apresentado na Figura 6.a, não é parâmetro editável no *script* para cálculo dos ganhos do controlador, e não pôde ser transferido ao usuário da *GUI*.

#### **3.1.5. Dados de saída da *GUI***

Os dados de saída são os ganhos de projeto para aplicação a cada compensador ressonante adicionado ao sistema. Ainda, o ganho de corrente e o ganho do erro na malha de realimentação do sistema são calculados e apresentados na saída do sistema.

O número mínimo de valores de saída é quatro, sendo estes o valor do ganho de corrente, o ganho do erro na malha de realimentação, e os dois parâmetros de ganho do controlador ressonante, para a opção de apenas um compensador ressonante. O número máximo de valores de saída é catorze, sendo novamente os dois primeiros relativos ao ganho de corrente e erro, respectivamente, e os demais relativos aos ganhos dos seis compensadores ressonantes adicionados ao sistema.

É necessário o uso de uma matriz 6x2 para exibir os ganhos dos 6 compensadores ressonantes que podem ser adicionados, além de mais duas saídas de dados para o ganho de corrente e erro do sistema.

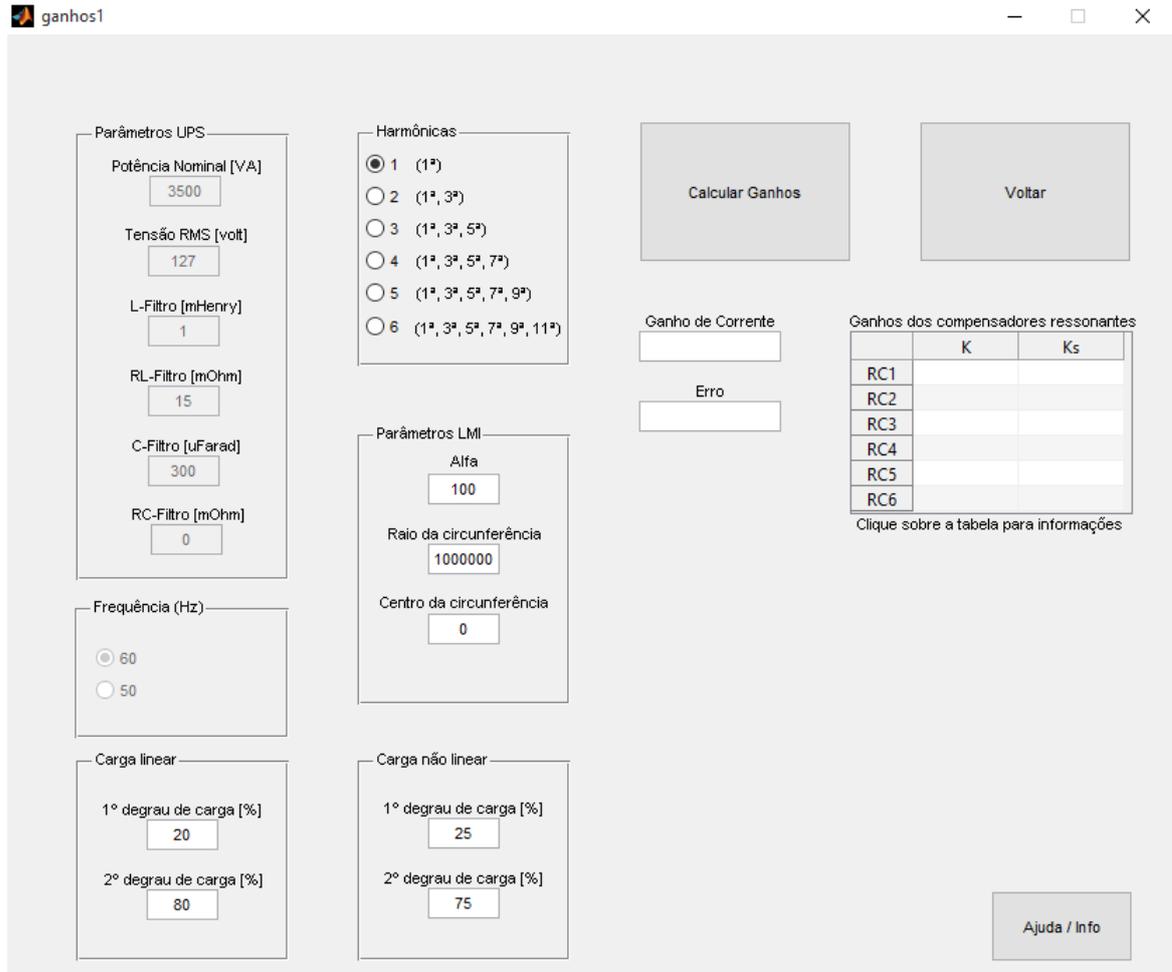
### 3.2. INTERFACE GRÁFICA PROPOSTA

A interface deve apresentar os dados de interesse de forma organizada e em um mesmo ambiente. A Figura 11 apresenta a interface para entrada de dados e saída dos ganhos dos compensadores ressonantes, ganho de corrente e erro.

Devido à utilização da “*GUI guide*” do MatLab® para criação da *GUI* da Figura 11, o cabeçalho do código da mesma com cada um dos blocos apresentados é criado automaticamente. Ainda, percebe-se que todos os blocos de entrada de dados estão previamente preenchidos. Os dados utilizados são referentes à *UPS* utilizada para os ensaios práticos, mas podem ser modificados para se obter os ganhos referentes a qualquer outra *UPS* que possua etapa CC-CA com modelo igual ao utilizado neste trabalho.

O objetivo é apresentar uma interface simples, que permita a edição dos dados relativos à *UPS*, parâmetros de carga, parâmetros do controlador e parâmetros acerca das *LMI* simultaneamente à exibição dos resultados associados a esses dados. Cada bloco apresentado na Figura 11 será explicado separadamente nas seções seguintes.

Figura 11. Interface gráfica principal



Fonte: Gerado pelo autor, através do software MatLab

### 3.3. CONJUNTO DE GUI

É necessário organizar o programa como um todo, para a utilização do mesmo ser intuitiva. Assim, se optou por colocar uma “janela de abertura”, que disponibiliza opções iniciais ao usuário. Esse programa apresenta ao usuário apenas opções básicas, sendo que entre elas haverá a opção de chamar a interface gráfica cuja finalidade é o cálculo dos ganhos para o controlador ressonante.

A interface gráfica principal, apresentada na Figura 11, é a parte do programa que permite o projeto do controlador e poderá ser iniciada a partir da tela de abertura. Tal interface apresenta a opção de voltar a janela de abertura, bem como oferece a possibilidade de iniciar a terceira e a quarta GUI, com informações sobre o uso e significado das caixas de entrada e saída de dados.

Após o cálculo dos ganhos dos compensadores ressonantes, uma quinta *GUI* é apresentada ao usuário. Essa possui apenas uma questão e dois botões. Um deles permite ao usuário executar uma simulação da *UPS* em funcionamento e visualizar a forma de onda da saída e a tensão RMS esperadas para o controlador ressonante calculado.

As seis *GUI*'s receberam as seguintes denominações:

“ABRIR1” – tela de abertura;

“GANHOS1” – interface para cálculo dos ganhos;

“help1” – Interface de ajuda sobre dados da tabela de saída;

“help2” – Interface de ajuda sobre dados de entrada da *GUI*;

“FECHAR” – interface com opção de executar ou não a simulação;

“Gui\_Graf” – apresenta forma de tensão da saída do sistema e seu valor RMS, caso tenha se executado a simulação.

### 3.3.1. Interface de abertura

Esta é a primeira interface gráfica do programa. Contém apenas três botões, com as funções de abrir a interface gráfica principal (com duas configurações diferentes) ou fechar a própria interface de abertura. Pode-se ver a imagem desta tela na Figura 12.

Figura 12. Interface de abertura



Fonte: Gerado pelo autor, através do software MatLab

“*UPS Padrão*”

Através deste *pushbutton*, inicia-se a interface principal com os valores da *UPS* disponível para ensaios. A *UPS* correspondente a estes dados está disponível fisicamente para testes nesta Universidade e é possível eventualmente comparar os dados obtidos pela interface com os respectivos resultados práticos.

O *pushbutton* possui *call-back* de função, conforme referido anteriormente. Quando pressionado, o botão “*UPS Padrão*” escreve o dado “1” em uma variável chamada “abrir”. Logo após definir o valor da variável “abrir”, o programa executa a *GUI* chamada GANHOS1, nome da interface principal. Finalmente, o programa fecha a própria janela da *GUI* “ABRIR1”.

“*UPS em branco*”

Através deste *pushbutton*, inicia-se a interface principal com os dados da *UPS* editáveis. Quando pressionado o *pushbutton* “*UPS em branco*”, o programa escreve “2” na variável “abrir”, e então executa a *GUI* com a interface principal, “GANHOS1”. Após, o programa fecha a própria janela da *GUI* “ABRIR1”.

“Sair”

Este *pushbutton* fecha o programa.

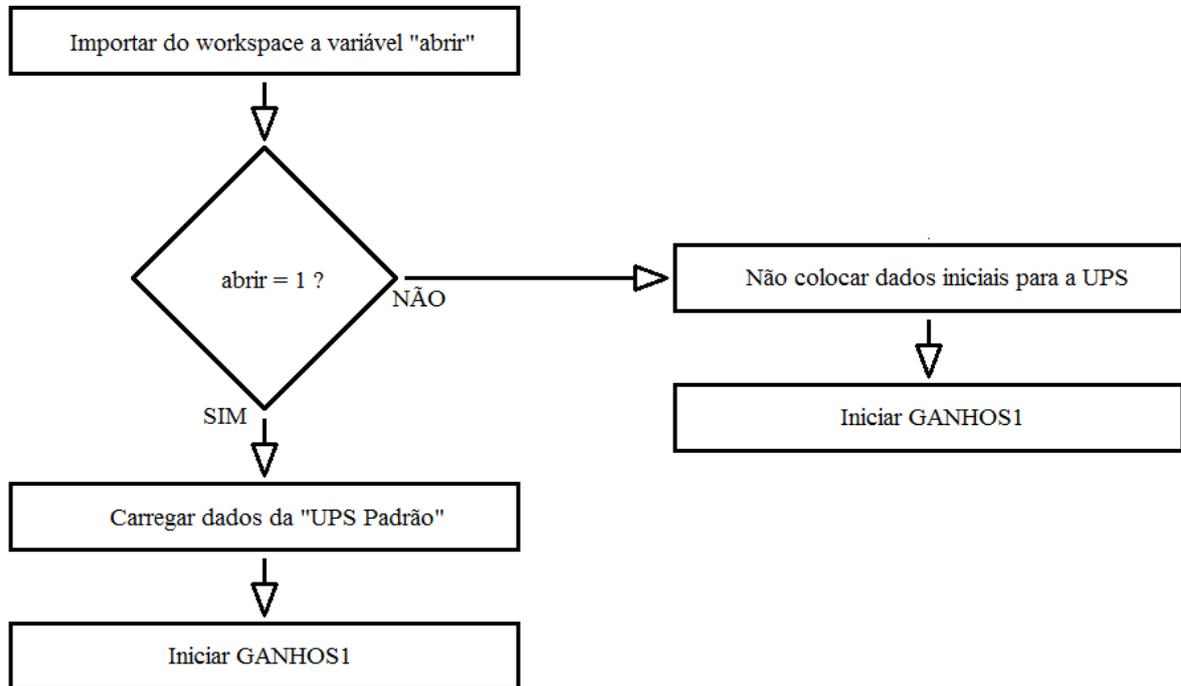
### 3.3.2. Interface para cálculo dos ganhos

Apresentada na Figura 11, esta é a interface principal do programa. Possui uma série de possibilidades de entrada de dados e apresenta os ganhos de corrente, o erro e os ganhos dos compensadores ressonantes.

Como apresentado anteriormente, esta interface pode ser apresentada com os dados da “*UPS Padrão*” (*UPS* cujos dados correspondem à da *UPS* disponível para testes) ou com as caixas dos dados da *UPS* em branco. Essa seleção é feita na interface gráfica de abertura (‘ABRIR1’), e coloca no *workspace* do MatLab® uma variável com o dado ‘1’ ou ‘2’.

Assim que a interface principal é aberta, verifica-se se o usuário gostaria de utilizar os parâmetros da *UPS* pré-definidos ou não (lendo `abrir`). Essa seleção é realizada conforme o fluxograma da Figura 13.

Figura 13. Carregar dados iniciais da *UPS* na interface principal



Fonte: Gerado pelo autor

As possibilidades de entrada de dados são organizadas em grupos. Estas são discutidas grupo a grupo nos próximos parágrafos. Os blocos de saída de dados e os botões serão explicados individualmente.

#### “Parâmetros *UPS*”

São os parâmetros de saída básicos da *UPS*. Dado que este programa é funcional apenas para a *UPS* meia-ponte descrita no capítulo 2.1, os parâmetros que definem o conversor CC-CA da *UPS* são resumidos nos itens abaixo:

Potência Nominal [VA] – potência nominal da *UPS*, dada em VA, conforme seção anterior. A entrada se dá através de um clique sobre a caixa de texto correspondente e digitação do valor. Para completar a entrada do dado, basta clicar fora da caixa de texto ou pressionar ENTER no teclado.

Tensão RMS [Volt] – É a tensão RMS de saída da *UPS*. Novamente, trata-se de uma caixa de texto, bastando clicar sobre a mesma, digitar o valor desejado, e clicar em outro ponto da *GUI*, fora da caixa de texto.

L-Filtro [mHenry] – A indutância do filtro de saída (conforme Figura 4), dada em mili Henry.

RL-Filtro [mOhm] – A resistência elétrica em série com o indutor do filtro de saída (conforme Figura 4), dada em mili Ohm. Seria conveniente utilizar na *GUI* o símbolo para o caractere Ohm ( $\Omega$ ), pois trata-se de um símbolo bastante difundido como unidade de resistência elétrica. Porém, a aplicação “*gui-guide*” do MatLab® não permite caracteres especiais.

C-Filtro [mHenry] – A capacitância do filtro de saída (conforme Figura 4), dada em micro Farad.

RC-Filtro [mOhm] – A resistência elétrica em série com o capacitor do filtro de saída, dada em micro Farad. Na prática, este valor é sempre muito próximo de zero, e é considerado nulo para a *UPS* Padrão.

“Frequência”

É a frequência de saída do sinal de tensão do conversor CC-CA. Na prática, utiliza-se o valor de 60Hz por padrão, em acordo com o padrão do sistema elétrico brasileiro. Adiciona-se tal parâmetro à *GUI* devido aos diferentes padrões de frequência da rede elétrica existentes ao redor do mundo, comumente 50Hz ou 60Hz. Um grupo de botões tipo *radio-buttons* faz com que as opções sejam mutuamente exclusivas.

“Carga linear”

1º degrau de carga: valor, em % (por cento) da potência nominal de saída, para o primeiro degrau de carga o qual a *UPS* será submetida na simulação (caso o usuário opte por executar simulação).

2º degrau de carga – Após a aplicação do primeiro degrau de carga, se aplica uma nova carga na saída da *UPS*. A nova potência exigida pela carga é dada pelo segundo degrau

de carga. Novamente, o valor dessa variável é dado em porcentagem em relação à potência nominal da *UPS*.

#### “Carga não linear”

1º degrau de carga: da mesma forma que ocorre com a carga linear, os requisitos de desempenho devem ser atendidos para dois degraus de carga não linear. A carga aqui em questão é o retificador de ponte completa de diodos, apresentado na Figura 1, e a unidade é por cento em relação à potência nominal.

2º degrau de carga – como ocorre com a carga linear, se aplica um novo valor de carga após a aplicação do primeiro degrau. A nova potência dissipada pela carga é dada pelo segundo degrau de carga.

#### “Harmônicas”

O compensador harmônico é adequado para rejeitar o erro especialmente para cargas não lineares com comportamento periódico cujas harmônicas em que a carga tenha maior energia associada são conhecidas. Para uma carga resistiva ideal, seria suficiente aplicar um compensador harmônico para a frequência fundamental (para seguimento em regime permanente).

Aqui é possível escolher em quantas harmônicas da frequência fundamental serão adicionados compensadores harmônicos. A aplicação de apenas um compensador ressonante influencia apenas na frequência fundamental.

Como existe um número limitado de opções para este dado de entrada, aplica-se um grupo de botões mutuamente exclusivos que contém as possibilidades previstas. Conforme [3], para cargas não lineares padrão, existe, sobretudo, efeito até a décima primeira harmônica.

#### “Parâmetros *LMF*”

Este grupo de caixas de texto permite a edição dos parâmetros ‘alfa’, ‘raio da circunferência’ e ‘centro da circunferência’.

### “Calcular Ganhos”

É o *pushbutton* que executa a captura dos dados da *GUI* principal e executa o código “roda\_teste” para resolver o problema dos ganhos do controlador. Primeiramente, capturam-se todos os dados de entrada da *GUI* salvando-os em variáveis, no formato ‘*String*’.

Dados no formato ‘*String*’ são dados com caracteres quaisquer da tabela ASCII [14], que podem ser inseridos em qualquer caixa de “*edit text*” ao longo da *GUI* principal. Porém, os dados necessários para rodar o código “roda\_teste” são todos numéricos. Conseqüentemente, todos os dados capturados das caixas *edit text*, devem ser convertidos para dados numéricos. Isso pode ser realizado através da função *str2number* ou *str2double* [11].

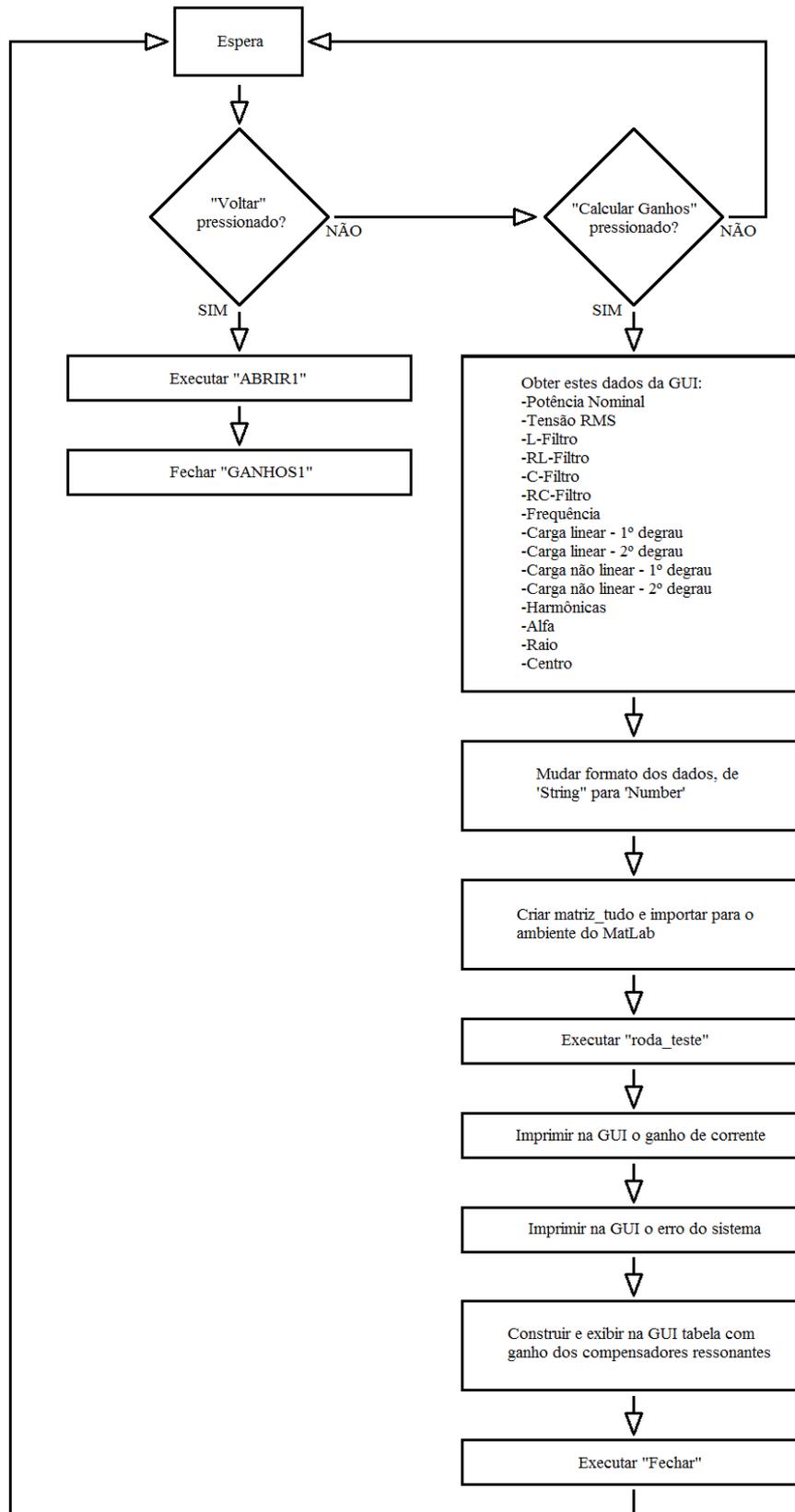
Este *pushbutton*, quando pressionado, executa um *call-back* que realiza a sequência mais importante do programa. Conseqüentemente, apresenta-se um fluxograma deste processo na Figura 14.

Conforme observa-se no fluxograma, existe um passo do programa em que os dados capturados da interface são exportados para o *workspace* do MatLab®. Isso se deve ao fato da simulação não ser possível sem esses dados disponíveis no *workspace* do programa.

### “Voltar”

Clicar nesse *pushbutton* fecha a *GUI* GANHOS1 e inicia a *GUI* ABRIR1, retornando o programa à tela de abertura.

Figura 14. Fluxograma do programa da GUI

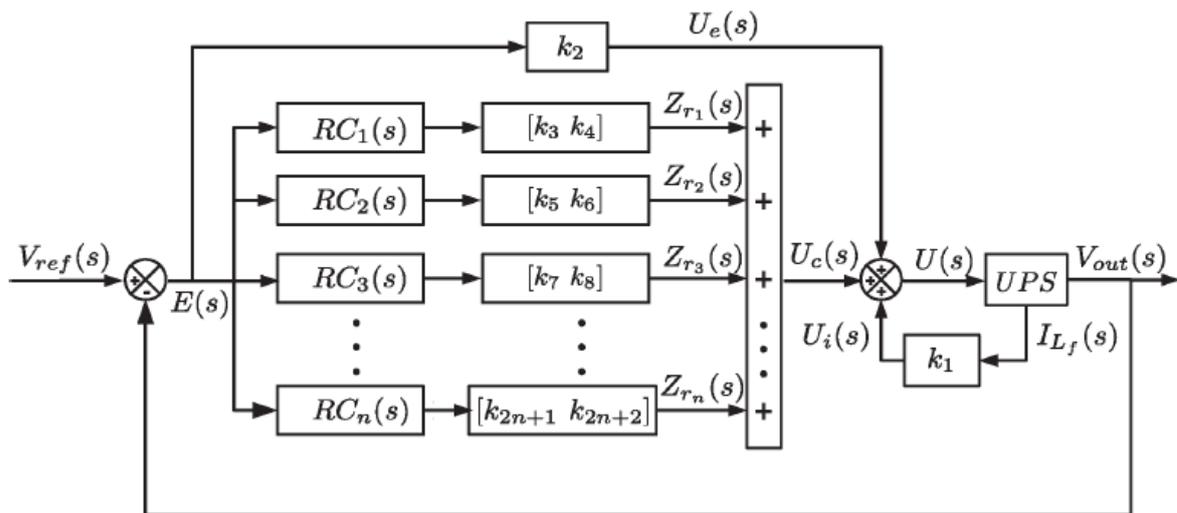


Fonte: Gerado pelo autor

“Ganho de Corrente” e “Erro”

O sistema da *UPS* controlada por compensadores ressonantes, apresentado na Figura 15, conforme [10], apresenta os ganhos de cada compensador ressonante ao lado do bloco do respectivo compensador ressonante (na figura, a sigla RC refere-se à *Ressonant Compensator*, tradução de compensador ressonante). Nessa figura, repara-se que os ganhos referentes ao primeiro compensador ressonante recebem os índices  $K_3$  e  $K_4$ .

Figura 15. Sistema completo do controlador ressonante



Fonte: [10]

Observando-se o sistema apresentado na Figura 15, se vê que o sinal de corrente  $I_{L_f}(s)$ , referente à corrente no indutor do filtro de saída da *UPS*, recebe um ganho antes de ser utilizado como sinal de realimentação da planta e é renomeado como  $U_i(s)$ . A corrente  $I_{L_f}(s)$ , ou seja, a corrente no indutor visto na Figura 4, é a própria corrente de saída da etapa CC-CA

Ainda, o sinal de erro  $E(s)$  é utilizado diretamente na planta, além de ser utilizado como sinal para os compensadores ressonantes. A parcela que é aplicada à planta recebe um ganho e é renomeada como  $U_e(s)$ .

Os ganhos de corrente e de erro são chamados de  $K_1$  e  $K_2$ , respectivamente, na Figura 15. Como apresentado na Figura 11, na interface gráfica principal essas duas constantes estão dispostas separadamente nas caixas de saída de texto sob os títulos “ganho de corrente” e “erro”. Dessa forma, a tabela com os ganhos tem apenas dados referentes aos compensadores ressonantes, simplificando a leitura.

“Ganhos dos compensadores ressonantes”

A tabela sob o título “ganho dos compensadores ressonantes”, na interface gráfica principal, apresentada na Figura 11, tem como dados de saída os ganhos de cada compensador aplicado ao sistema. A organização desses ganhos é dada conforme apresentado na Figura 15, onde ‘K3’ e ‘K4’ são referentes ao primeiro compensador ressonante, ‘K5’ e ‘K6’ referentes ao segundo compensador ressonante, assim sucessivamente. O termo ‘K’ par sempre multiplica a variável ‘s’, e a o termo ‘K’ ímpar sempre é independente.

### 3.3.3. Interface de ajuda para saída de dados

Conforme indicação presente na *GUI* principal pode-se clicar na tabela com os ganhos dos compensadores ressonantes a fim de obter informações adicionais. Um clique nesta tabela apresenta a *GUI* chamada “help1”, com informações sobre a saída da interface gráfica.

A interface gráfica “help1” é quase completamente passiva, tendo como único componente executando uma ação o botão “Fechar”, no canto inferior direito. O botão “Fechar” tem como função fechar a *GUI* “help1”.

A *GUI* “help1” pode ser vista na Figura 16 e basicamente resume a função dos dados importantes ao usuário.

### 3.3.4. Interface de ajuda para entrada de dados

No canto inferior direito da *GUI* principal, existe um *pushbutton* “Ajuda/Info”. Clicando no mesmo, apresenta-se a interface com informações sobre os parâmetros de entrada. Esta interface recebe o nome de “help2” e é apresentada na Figura 17.

Figura 16. Interface help1

help1

Na tabela "Ganho dos compensadores ressonantes", apresentada na interface gráfica (repetida à direita), cada linha da tabela se refere a um único compensador ressonante. Cada compensador ressonante possui DOIS parâmetros constantes, que estão nas duas colunas da tabela.

A fim de facilitar a compreensão, a tabela à direita tem em cada célula um número, representando sua posição dentro dos compensadores ressonantes, apresentados na figura abaixo.

Na interface gráfica, se escolhe quantas harmônicas se deseja compensar. Esse número define na verdade, quantos compensadores serão aplicados ao sistema.

Ganhos dos compensadores ressonantes		
	K	Ks
RC1	K0	K1
RC2	K2	K3
RC3	K4	K5
RC4	K6	K7
RC5	K8	K9
RC6	K10	K11

Fechar

Fonte: Gerado pelo autor, através do software MatLab

Figura 17. Interface help2

help2

Parâmetros da UPS

A potência nominal e a tensão RMS são os principais parâmetros de saída da UPS. Dependem, sobretudo, dos componentes físicos que compõem a etapa inversora CC-CA da UPS.

Os componentes 'L-Filtro', 'RL-Filtro', 'C-Filtro' e 'RC-Filtro' são os componentes do filtro de saída da UPS, como se vê na Figura 1.

A Frequência de saída da UPS é a frequência da tensão de saída da mesma. Depende do sistema de chaveamento da etapa CC-CA, e pode assumir neste programa os valores 60Hz e 50Hz.

As variáveis que impõem os limites para as LMIs disponíveis para edição na interface gráfica estão representadas graficamente à esquerda, na Figura 2. Na imagem central, está representado o ângulo  $\Phi_{hi}$  ( $\varphi$ ), parâmetro este considerado 90°, sem possibilidade de modificação, aqui neste programa.

O cálculo dos ganhos dos compensadores ressonantes é feito de forma a melhor atender uma determinada faixa de carga. Os valores definidos nas caixas '1º degrau de carga [%]' e '2º degrau de carga [%]' indicam os saltos de carga que serão aplicados à saída da UPS.

Por exemplo, se forem aplicados os valores:  
 '1º degrau de carga [%] = 20'  
 '2º degrau de carga [%] = 80'

os ganhos obtidos serão otimizados para a situação em que a UPS funciona em aberto (sem carga), com um salto para 20% da carga nominal e um segundo salto, indo para 80% da carga nominal.

Haverá a otimização considerando tanto cargas lineares (resistivas) como para cargas não lineares (retificador com ponte de diodos, por exemplo).

Figura 1

Regiões LMI: (a) Semi-plano, (b) setor cônico, e (c) Disco.

Figura 2

Fechar

Fonte: Gerado pelo autor, através do software MatLab

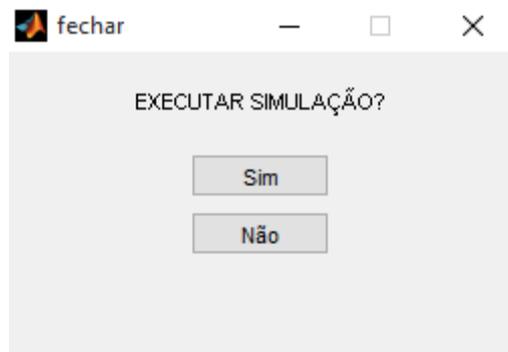
Tal interface possui cinco gráficos, que estão configurados para exibir uma figura específica cada. Ainda, no canto inferior direito, existe a opção de fechar a interface.

### 3.3.5. Interface com opção de simulação

Após o cálculo dos ganhos na interface principal, é exibida uma janela com a alternativa de executar ou não uma simulação. A Figura 18 apresenta a interface gráfica “FECHAR”, nome dado à interface em questão.

É apresentada a alternativa ao usuário devido ao tempo para simulação. Supõe-se que, normalmente, o interesse maior do usuário seja calcular os ganhos para aplicação no controlador da *UPS* em questão – sem a necessidade de executar uma simulação.

Figura 18. Interface gráfica “FECHAR”



Fonte: Gerado pelo autor, através do software MatLab

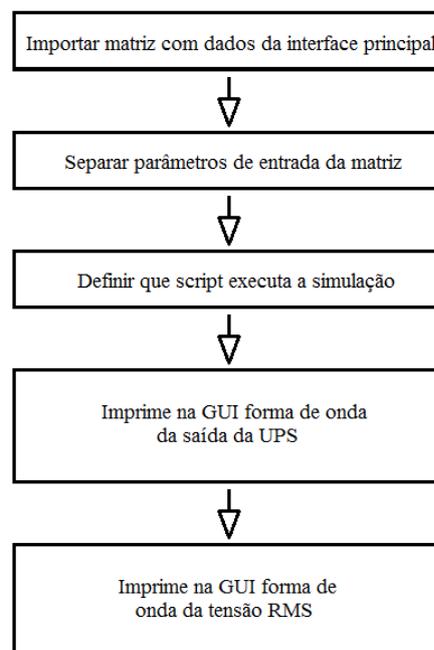
O programa executado pelo *call-back* do *pushbutton* “Sim” consiste em executar uma nova interface gráfica, chamada “Gui\_Graf”. O programa executado pelo *call-back* do *pushbutton* “Não” apenas encerra a própria *GUI* “FECHAR”.

### 3.3.6. Interface com tensão de saída da simulação

Caso o usuário opte por executar a simulação, uma nova interface gráfica é apresentada com a finalidade de exibir os gráficos de alguns parâmetros de saída da *UPS*. Os dados correspondem à forma de onda da saída da *UPS* e sua tensão RMS.

Ainda que esta tela não possua itens ativos (caixas de entrada/saída de texto ou *pushbuttons*) como a interface principal, executa uma sequência de passos antes de ser exibida. Os dados capturados na interface principal são salvos em uma matriz linha, para utilização neste passo. A interação entre diferentes *GUI* do MatLab® é realizada através do *workspace*. Como visto no fluxograma da Figura 14, a interface principal salva um vetor com todos os dados necessários para possibilitar a execução da simulação. O fluxograma da Figura 19 descreve o procedimento.

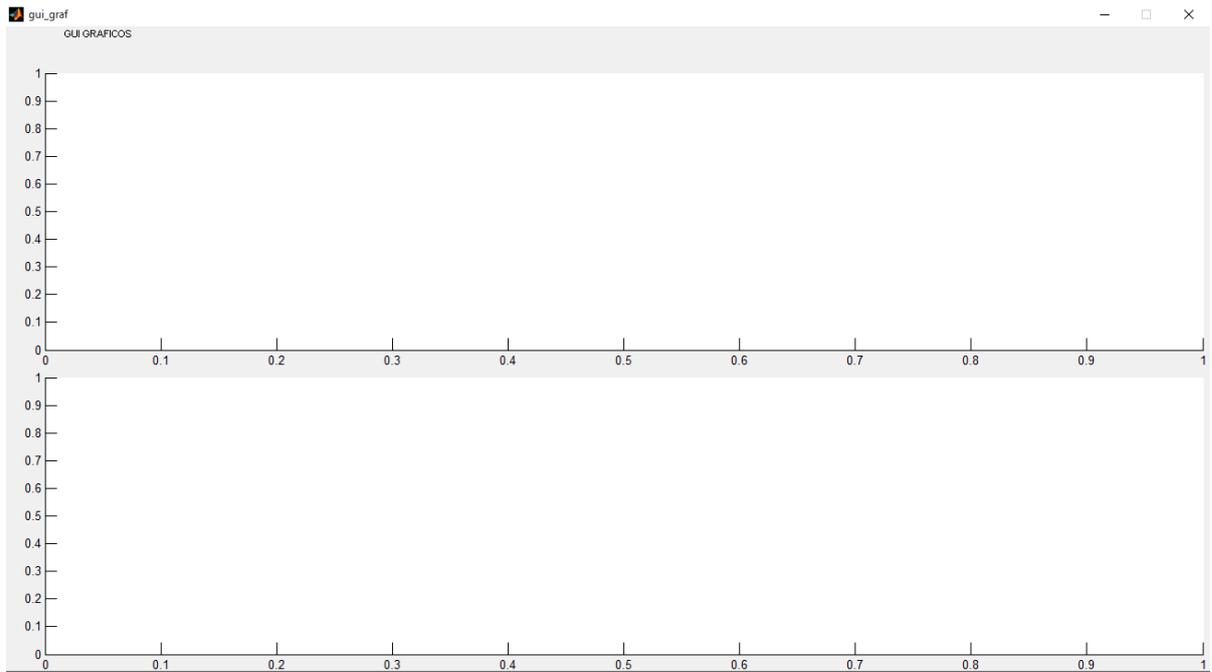
Figura 19. Fluxograma da *GUI* “Gui\_Graf”



Fonte: Gerado pelo autor

A interface gráfica é apresentada na Figura 20. O gráfico apresentado na parte superior da interface corresponde à forma de onda da saída da *UPS*, enquanto o gráfico apresentado na parte inferior representa a tensão RMS de saída.

Figura 20. Interface gráfica “Gui\_Graf”



Fonte: Gerado pelo autor, através do software MatLab

## 4. RESULTADOS

Foi apresentado no capítulo 3 como foi construída cada uma das interfaces gráficas propostas e os dados de interesse solicitados na interface principal. Para tal, se usou de fluxogramas e imagens das próprias interfaces.

Aqui, será apresentado o cálculo dos ganhos de um controlador ressonante com um único compensador ressonante, atuando na frequência fundamental de operação da *UPS*. Os dados da *UPS* aplicados à interface gráfica são baseados nos dados conhecidos de uma etapa inversora CC-CA de 3,5kVA monofásica. Será mostrado cada passo do processo, e se optará por executar a simulação. Um segundo controlador será projetado com o uso da interface, esse com dois compensadores ressonantes para aplicação na planta – mas para esse, não será executada a simulação (pretende-se apenas analisar o efeito na *UPS* real de se adicionar um compensador ressonante).

Finalmente, serão apresentados resultados obtidos em laboratório com a *UPS* em questão, na qual o controlador é aplicado à etapa CC-CA através do uso de recursos de tempo real das plataformas MatLab® e Simulink®, em conjunto com uma placa DSP D-Space. Os dados de saída foram capturados com o uso do analisador de qualidade de energia Fluke43B Single-Phase Power Quality Analyser e seu software para comunicação com a plataforma Windows, bem como pelo osciloscópio digital Tektronics TDS 2014.

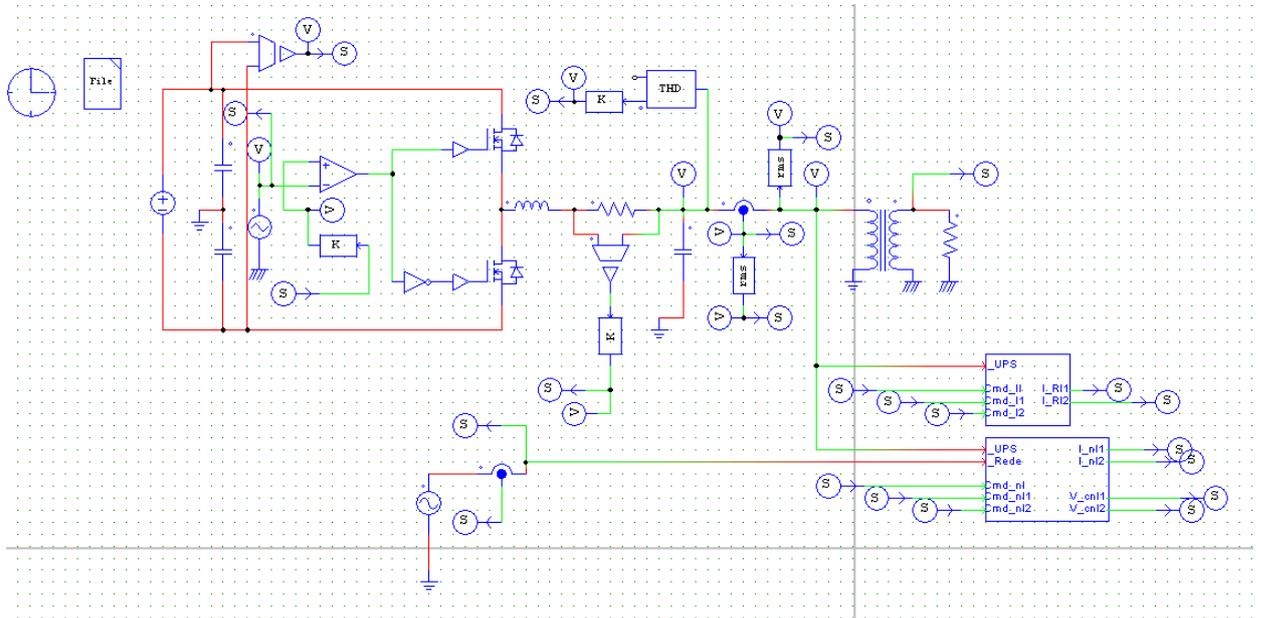
### 4.1. MÉTODO DE SIMULAÇÃO

A execução da simulação requer que exista recurso que simule a *UPS*, sendo esta, descrita pelos parâmetros dados na interface principal do programa. Aqui, um modelo em Powersim PSIM® foi utilizado, em conjunto com o Simulink® para simular o controlador. Ambas soluções são parte da proposta em [3]

Quando solicitada a execução da simulação, os dados necessários para execução da mesma (parâmetros de entrada relacionados à *UPS*, bem como o controlador calculado) deverão estar no *workspace* do MatLab®, visto que não é possível fazer a comunicação entre o *script* da *GUI* e o PSIM® diretamente. Todos os dispositivos de medição utilizados no modelo da *UPS* dentro do PSIM® poderão ser utilizados para obter dados e executar *plot* de gráficos na interface do MatLab®.

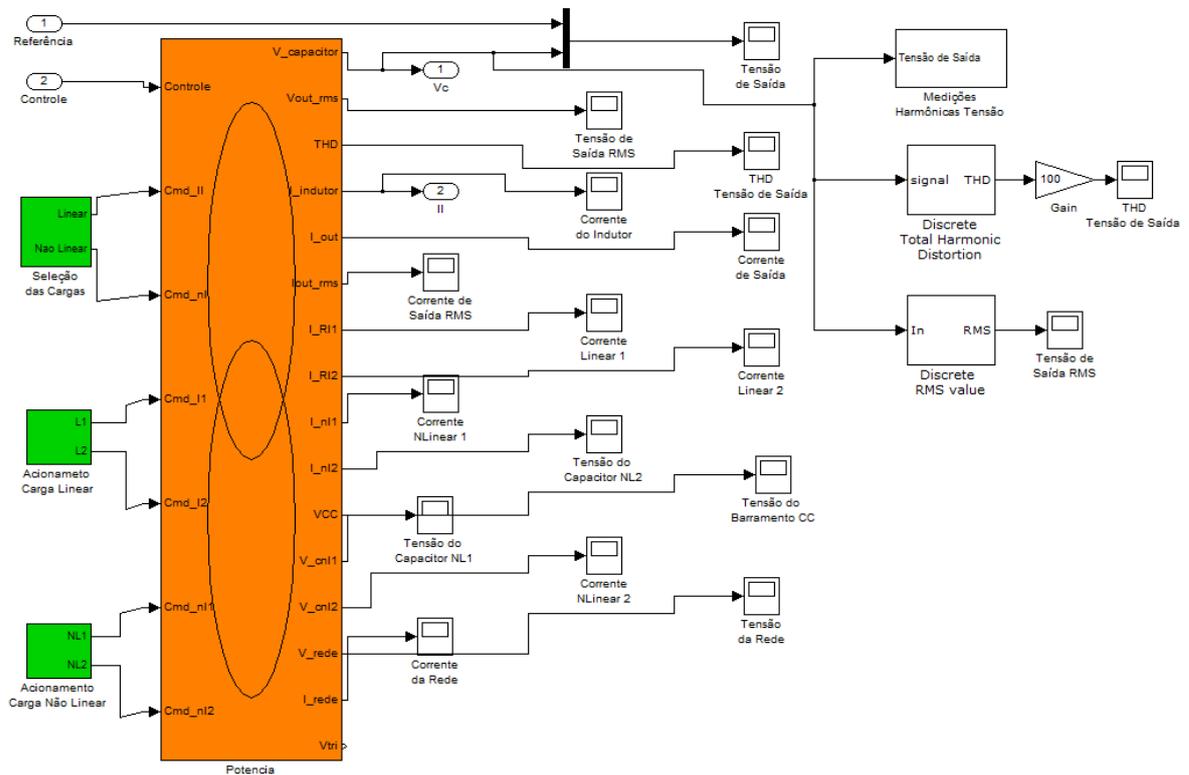
Na Figura 21, é apresentado o modelo para a UPS no PSIM®, utilizado para a simulação. A Figura 22 apresenta o modelo no Simulink®.

Figura 21. Modelo do conversor CC-CA da UPS, no PSIM



Fonte: Adaptado de [3]

Figura 22. Modelo do sistema no Simulink, com bloco de ligação (em laranja) para comunicação com o PSIM



Fonte: Adaptado de [3]

O modelo da *UPS* e o diagrama no Simulink® são parte de [3]. Na Figura 22, integração entre os *softwares* é feita pelo SimCoupler, bloco em laranja. Os blocos na cor verde são responsáveis pela simulação da inserção de carga.

#### 4.2. CÁLCULO DOS GANHOS PARA *UPS*

A etapa CC-CA da *UPS* cujos ganhos do controlador podem ser obtidos por meio do programa proposto nesse trabalho, foi apresentada na Figura 4, e trata-se de uma etapa inversora pela topologia meia-ponte. A unidade disponível no Laboratório do GCAR tem os parâmetros apresentados na Tabela 1.

Tabela 1. Parâmetros da *UPS* para testes

Parâmetro	Valor
Potência Nominal	3,5kVA
Tensão de saída RMS	127V
L-filtro	0,001H
RL-filtro	0,015Ω
C-filtro	300μF
RC-filtro	0Ω

Fonte: Dados da *UPS* real, utilizada para testes

A frequência da tensão de saída da *UPS* é a padrão do Brasil, 60Hz, de acordo com a norma NBR5410 [15], e seguindo as normas IEEE 944 Std [16], e IEC 62040 [1], para a faixa de potência desta *UPS*, os ensaios devem ser realizados, na sequência descrita: com 0% da potência nominal, 20% da potência nominal, 80% da potência nominal, 20% da potência nominal e 0% da potência nominal para cargas lineares e, para cargas não lineares, com: 0% da potência nominal, 25% da potência nominal, 75% da potência nominal, 25% da potência nominal e 0% da potência nominal.

Os parâmetros *LMI* devem ser escolhidos pelo usuário, conforme necessidade de velocidade de resposta e resposta em regime permanente. Os parâmetros aqui escolhidos são apresentados na Tabela 2.

Tabela 2. Parâmetros *LMI* utilizados para projeto do controlador

Parâmetro	Valor
$\alpha$ (posição Alfa)	100
r (raio da circunferência)	1000000
c (centro da circunferência)	0

Fonte: Dados utilizados em ensaios apresentados em [3]

Para o controlador do ensaio, se optou pela aplicação de um compensador ressonante, colocado na frequência fundamental de resposta do sinal de referência – conforme explicado. Dessa forma, têm-se todos os dados de entrada necessários para a obtenção dos ganhos do controlador.

Abrindo janela de abertura, se seleciona a opção “*UPS em branco*”, destacada na Figura 23.

Figura 23. Primeiro passo para obter ganhos do controlador



Fonte: Gerado pelo autor, através do software MatLab

Na interface principal, se preenche os dados da *UPS*, degraus de carga, quantidade de compensadores ressonantes e os parâmetros *LMI*, conforme Figura 24. Em seguida, se clica em “Calcular Ganhos”, indicado na Figura 25.

Figura 24. Parâmetros de entrada, já aplicados à GUI

Parâmetros UPS

Potência Nominal [VA]  
3500

Tensão RMS [volt]  
127

L-Filtro [mHenry]  
1

RL-Filtro [mOhm]  
15

C-Filtro [uFarad]  
300

RC-Filtro [mOhm]  
0

Harmônicas

1 (1ª)

2 (1ª, 3ª)

3 (1ª, 3ª, 5ª)

4 (1ª, 3ª, 5ª, 7ª)

5 (1ª, 3ª, 5ª, 7ª, 9ª)

6 (1ª, 3ª, 5ª, 7ª, 9ª, 11ª)

Parâmetros LMI

Alfa  
100

Raio da circunferência  
1000000

Centro da circunferência  
0

Frequência (Hz)

60

50

Carga linear

1º degrau de carga [%]  
20

2º degrau de carga [%]  
80

Carga não linear

1º degrau de carga [%]  
25

2º degrau de carga [%]  
75

Calcular Ganhos

Voltar

Ganho de Corrente  
[ ]

Erro  
[ ]

Ganhos dos compensadores ressonantes

	K	Ks
RC1		
RC2		
RC3		
RC4		
RC5		
RC6		

Clique sobre a tabela para informações

Ajuda / Info

Fonte: Gerado pelo autor, através do software MatLab

Figura 25. Indicação de *pushbutton* para executar cálculo dos ganhos

Calcular Ganhos

Voltar

Ganho de Corrente  
[ ]

Erro  
[ ]

Ganhos dos compensadores ressonantes

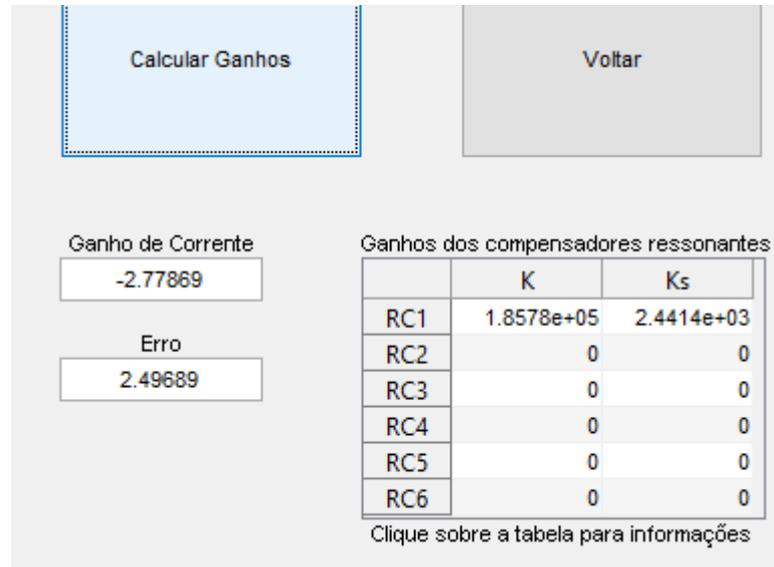
	K	Ks
RC1		
RC2		
RC3		
RC4		
RC5		
RC6		

Clique sobre a tabela para informações

Fonte: Gerado pelo autor, através do software MatLab

Após clicar no *pushbutton* “Calcular Ganhos”, as caixas de dados de saída e a tabela com os ganhos dos compensadores ressonantes são atualizadas. Os valores obtidos estão apresentados, da maneira que é apresentado na interface, na Figura 26.

Figura 26. Resultados para a *UPS* de interesse, com um compensador ressonante



Fonte: Gerado pelo autor, através do software MatLab

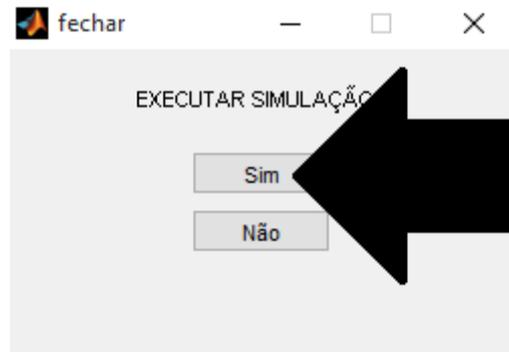
Considerando a Figura 15, os termos K1, K2, K3 e K4 estão dispostos na Tabela 3. A interface gráfica com a alternativa de executar a simulação será exibida. Conforme indicado na Figura 27, o usuário deve clicar em “Sim”.

Tabela 3. Ganhos do controlador com um compensador ressonante

Parâmetro	Valor
K1	$-2.77869$
K2	$2.49689$
K3	$1.85781e+05$
K4	$2.4414e+03$

Fonte: Gerados pelo autor, com o uso da GUI

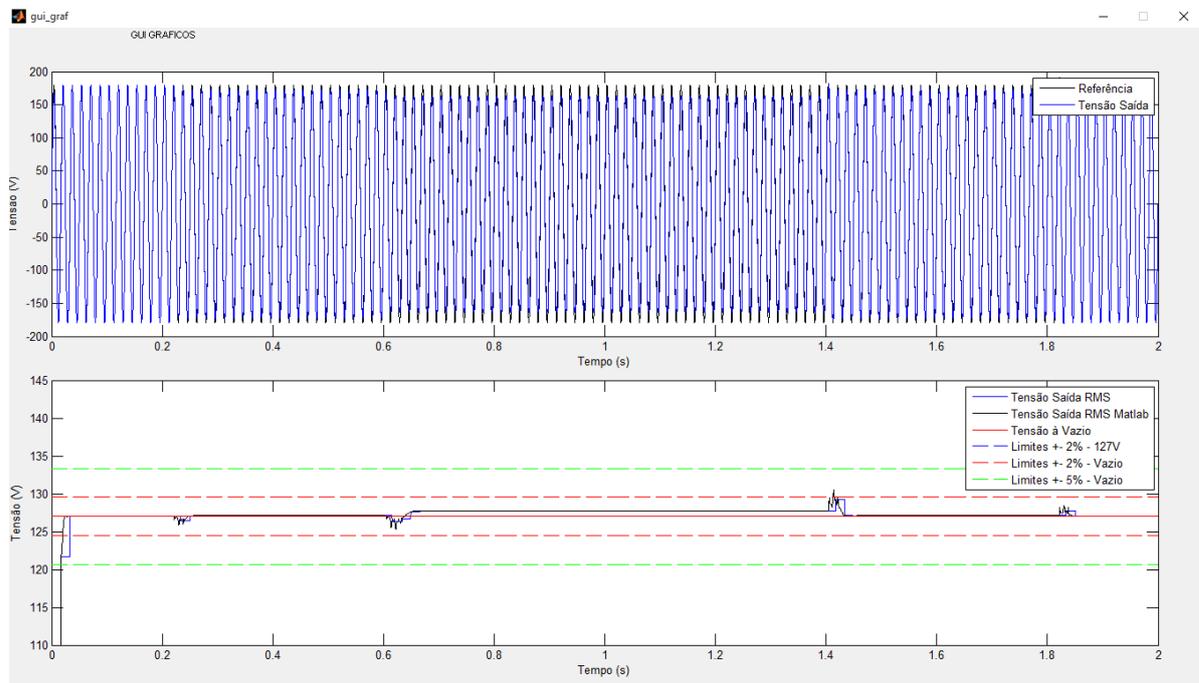
Figura 27. Solicitar a execução da simulação



Fonte: Gerado pelo autor, através do software MatLab

Após a simulação terminar, a interface gráfica com os gráficos da tensão de saída e tensão RMS de saída será apresentada. Para os dados aqui utilizados, os resultados estão apresentados na Figura 28.

Figura 28. Gráficos da tensão de saída e da tensão RMS de saída na GUI



Fonte: Gerado pelo autor, através do software MatLab

Observa-se na Figura 28 que a tensão RMS de saída para a carga máxima é 128 V. Esse é o dado de maior interesse, e será comparado com os dados do ensaio prático na sequência. A tensão para o sistema sem carga ( $t < 0,2s$ ) é aproximadamente 127,5V.

Para fins de analisar as componentes harmônicas da tensão de saída, se realizou um novo cálculo de ganhos, para a mesma *UPS*, utilizando compensadores harmônicos para a primeira e a terceira componentes. Os resultados estão apresentados na Tabela 4.

Tabela 4. Ganhos do controlador com dois compensadores ressonantes

Parâmetro	Valor
K1	-3.84742
K2	3.34125
K3	1.0071e+05
K4	3.0542e+03
K5	1.0853e+06
K6	2.8286e+03

Fonte: Gerados pelo autor, com o uso da GUI

#### 4.3. IMPLEMENTAÇÃO DO CONTROLADOR E RESULTADOS

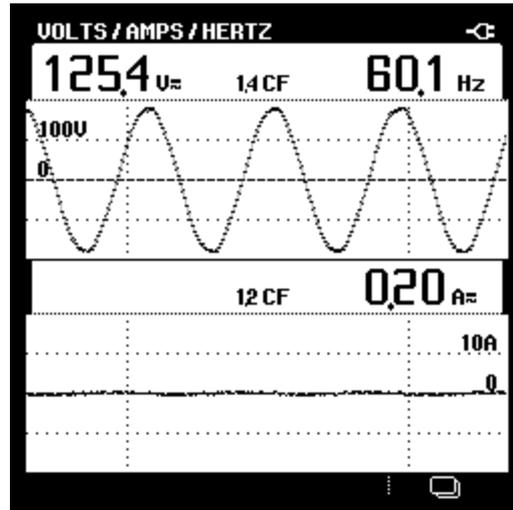
A expressão apresentada na Equação 1 é a forma de um compensador ressonante contínuo no domínio da frequência. Os valores das duas constantes associadas a cada compensador, bem como os ganhos de corrente e erro, acabam por gerar uma expressão contínua na frequência. Para ser possível utilizar as plataformas MatLab® e Simulink® e a placa de comunicação D-Space, será necessário obter a expressão discreta da função de transferência do controlador.

A forma utilizada foi a discretização por Tustin (ou bilinear), apresentada em [2]. O processo foi feito com o uso de *script* do MatLab®, proposto por [3], parte do código ‘roda\_teste’.

A aplicação de carga se dará de forma controlada também pelo MatLab/Simulink atuando em conjunto com o D-Space. O método utilizado para inserção de carga é o proposto em [12].

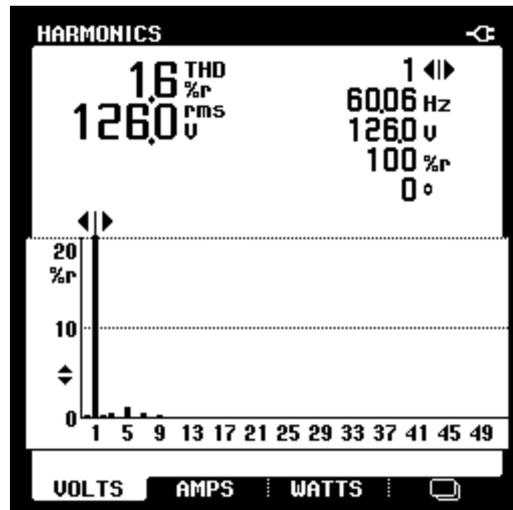
Se observa na Figura 29 a forma de onda da saída e a corrente de saída com o sistema à vazio (sem carga), e na Figura 30 a THD da saída da *UPS*.

Figura 29. Forma de onda da tensão e corrente de saída da UPS, 0% de carga linear, um compensador ressonante



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 30. THD da saída da UPS, 0% de carga linear, um compensador ressonante

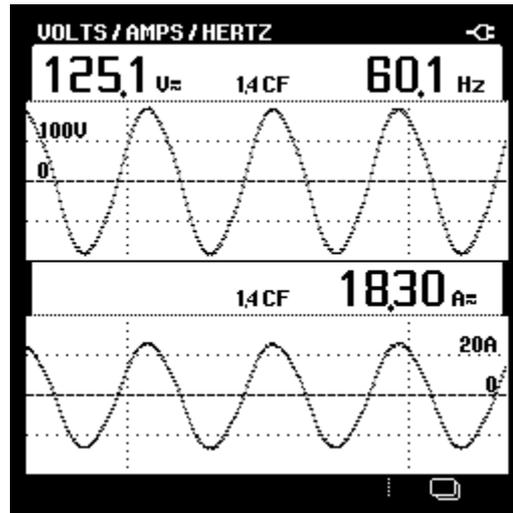


Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Observa-se tensão de 125,4V para o sistema à vazio. A saída apresenta praticamente toda energia na frequência fundamental do sistema.

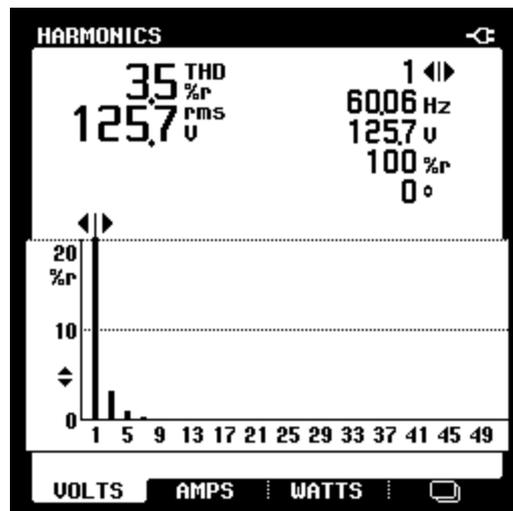
A Figura 31 apresenta a resposta para o sistema com 80% da carga nominal, carga puramente resistiva. A Figura 32 apresenta a THD do sistema funcionando com 80% da carga nominal, com carga puramente resistiva.

Figura 31. Forma de onda da tensão e corrente de saída da *UPS*, 80% de carga linear, um compensador ressonante



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 32. THD da saída da *UPS*, 80% de carga linear, um compensador ressonante



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

A tensão de saída é, nesse caso, 125,7V. A distorção harmônica total é 3,5%. Os dados comparando a tensão de saída com a simulação e com o ensaio estão apresentados na Tabela 5. Nessa, se observa concordância entre os valores esperados e observados.

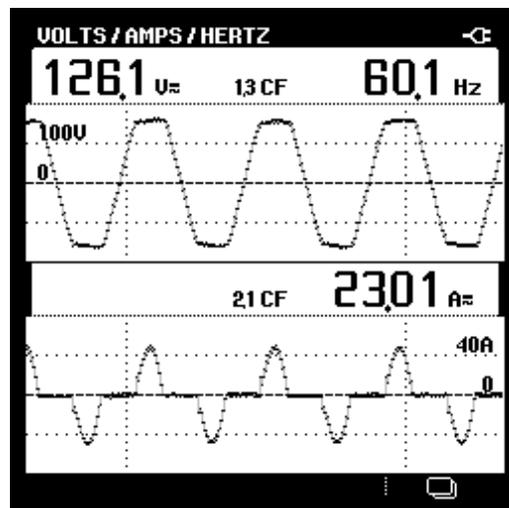
Tabela 5. Resultados para simulação/ensaio com carga linear

	Simulação	Ensaio
Vout (0% carga)	127,5Vrms	125,4Vrms
Vout (80% carga)	128Vrms	125,7Vrms

Fonte: Gerados pelo autor, com o instrumento osciloscópio digital Tektronics TDS 2014 e simulação

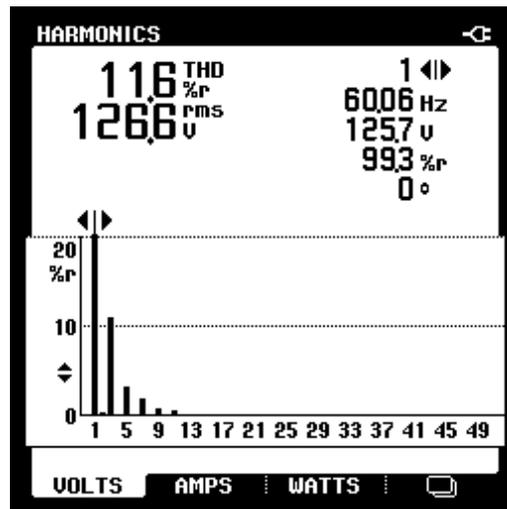
Foi realizado um ensaio com 80% da carga não-linear aplicado à saída da *UPS*. Este resultado, entretanto, não poderá ser comparado com os resultados da simulação, pois a *GUI* não imprime na saída gráficos para a resposta à carga não-linear. Na Figura 33, é apresentada a forma de onda da saída com 80% da carga não-linear, e na Figura 34, é exibida a THD para essa carga.

Figura 33. Forma de onda da tensão e corrente de saída da *UPS*, 75% de carga não-linear, um compensador ressonante



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 34. THD da saída da UPS, 75% de carga não-linear, um compensador ressonante

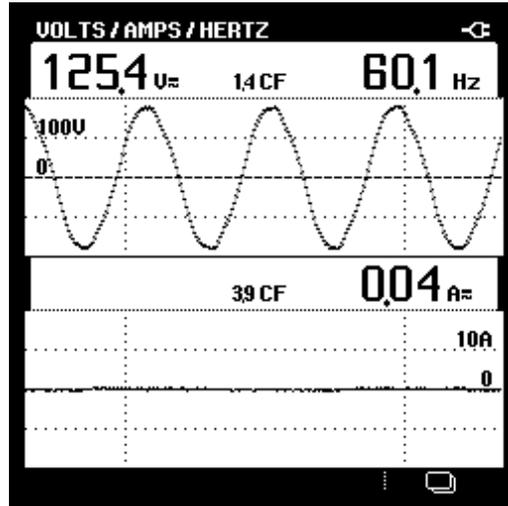


Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Observa-se que, com o uso de compensador ressonante com apenas um modo, atuando na frequência fundamental do sistema, quando aplicada carga não-linear, a componente da saída em terceira e quinta harmônicas sobem significativamente. O comportamento era esperado, visto que os ganhos foram calculados para cargas com componentes de corrente apenas na frequência fundamental.

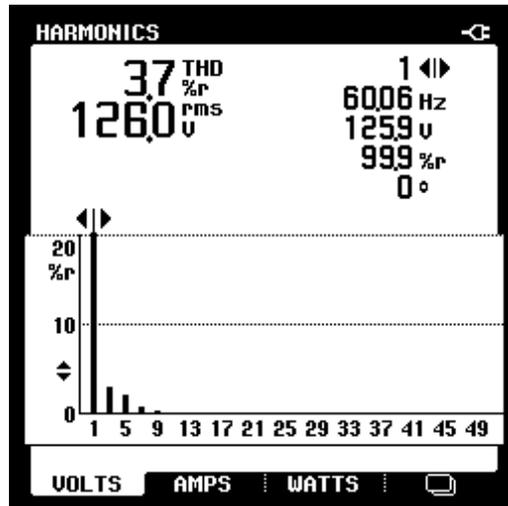
Ainda, um ensaio com o uso de dois compensadores ressonantes foi realizado, com os dados apresentados da Tabela 4. Foi realizado ensaio a vazio, com 80% da carga linear e 75% de carga não-linear. Para o ensaio a vazio, é apresentada a forma de onda da tensão e da corrente na Figura 35 e a THD na Figura 36.

Figura 35 Forma de onda da tensão e corrente de saída da *UPS*, 0% de carga linear, dois compensadores ressonantes



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

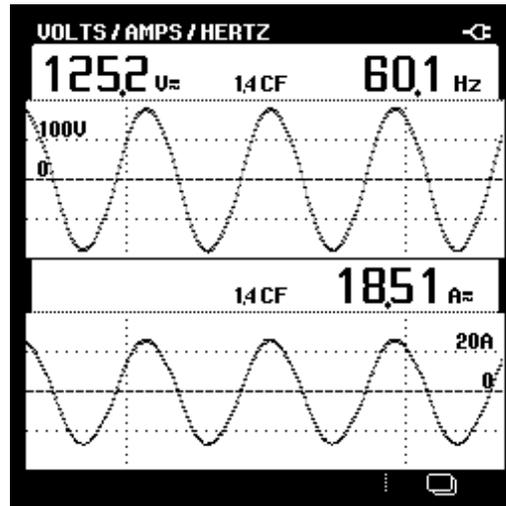
Figura 36. THD da saída da *UPS*, 0% de carga linear, dois compensadores ressonantes



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

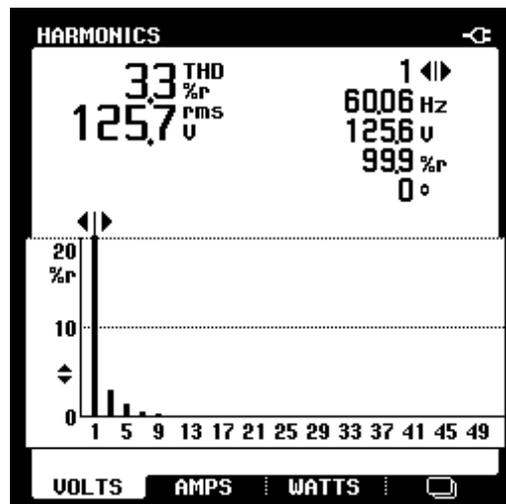
Para o ensaio com 80% da carga linear, é apresentada a forma de onda da tensão e da corrente na Figura 37 e a THD na Figura 38. Finalmente, para o ensaio com 75% de carga não-linear, a Figura 39 apresenta a forma de onda da tensão e da corrente de saída da *UPS*, e a Figura 40 apresenta sua THD.

Figura 37 Forma de onda da tensão e corrente de saída da UPS, 80% de carga linear, dois compensadores ressonantes



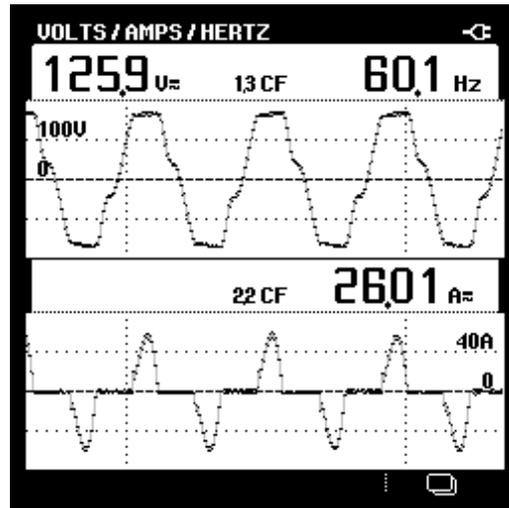
Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 38. THD da saída da UPS, 80% de carga linear, dois compensadores ressonantes



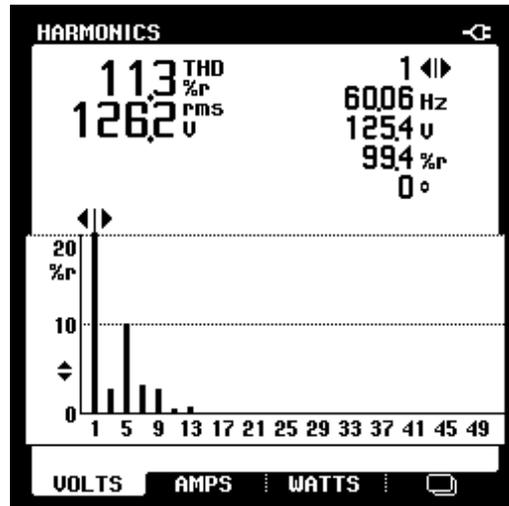
Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 39. Forma de onda da tensão e corrente de saída da *UPS*, 75% de carga não-linear, dois compensadores ressonantes



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Figura 40. THD da saída da *UPS*, 75% de carga não-linear, dois compensadores ressonantes



Fonte: Dados obtidos pelo autor através do instrumento Fluke 43B Single-Phase Power Quality Analyser

Comparando a Figura 32 e 4.17, observa-se que a resposta do controlador ressonante com dois compensadores ressonantes não é significativamente diferente da resposta do controlador utilizando apenas um compensador ressonante. O perfil das THDs é muito semelhante. Isso se deve ao fato da corrente da carga linear ter aproximadamente a mesma forma de onda da tensão de saída da *UPS*, como se observa na Figura 31 e na Figura 37.

Entretanto, comparando a Figura 34 com a Figura 40, observa-se grande diferença na disposição dos componentes harmônicos. Ainda que o uso de dois compensadores ressonantes

tenha tido pouco efeito na THD (11,3% para dois compensadores contra 11.6% para um compensador), a presença de um compensador na terceira harmônica reduziu significativamente a parcela dessa frequência na THD.

## 5. CONCLUSÃO

As *UPS* são equipamentos utilizados nas mais diversas aplicações, com grande variedade de perfis de carga. A confiabilidade do equipamento deve ser mantida, e isso torna complexo o objetivo de obter um único controlador, com parâmetros estáticos, que mantenha a qualidade de energia para qualquer tipo de carga. Aqui, se desenvolveu um trabalho no sentido de tornar rápido o processo de se obter os parâmetros para um controlador ressonante, onde diversos dados da *UPS* são fornecidos, bem como o número de modos ressonantes (a quantidade de compensadores ressonantes da malha de controle).

A solução do problema matemático que permite obtenção dos parâmetros do controlador ressonante não é o tema deste trabalho, e foi já proposta em trabalhos anteriores. Tal solução estava disponível na forma de *script* do MatLab®, e deveria ser utilizada para resolver os problemas envolvendo as restrições *LMI* (*Linear Matrix Inequalities*), processo pelo qual se obtém os ganhos do controlador ressonante.

O método escolhido para otimizar o uso do *script* descrito acima foi o uso de uma interface gráfica do MatLab®. Dessa forma, se evita edição de diversos códigos para se alterar os dados de interesse da *UPS* e do controlador – todas essas edições são feitas em uma única interface, onde os dados são facilmente identificáveis. O objetivo de compor uma interface amigável ao usuário, contendo todos os dados necessários para o cálculo dos parâmetros do controlador ressonante foi atingido.

O *script* responsável pela solução para os ganhos do controlador foi editado em alguns pontos, buscando transformar todos os dados de entrada em variáveis, de forma a ser possível editar tais valores, a partir dos dados aplicados à *GUI*. Este mesmo *script* apresenta uma série de arquivos acessórios – alguns com dados que deveriam ser alterados pela interface gráfica, caso o usuário assim desejasse. Todos esses arquivos acabaram por sofrer edições, a fim de permitir que a interface tivesse a possibilidade de alterar todos os dados que fossem necessários.

Para executar o cálculo dos parâmetros do controlador ressonante, um botão foi adicionado à interface gráfica. Assim, quando o usuário tivesse com todos os dados já colocados na interface gráfica, estaria disponível a possibilidade de executar o *script* para cálculo dos ganhos do controlador. Ainda, após o cálculo dos parâmetros do controlador, é apresentada ao usuário a possibilidade de executar uma simulação, onde a mesma irá

apresentar a estimativa da forma de onda de saída do sistema, bem como sua tensão RMS. Tal funcionalidade depende de haver no sistema instalado o PSIM®, e que o mesmo esteja configurado para funcionar em conjuntos com o MatLab®. A simulação é baseada em modelo simplificado da *UPS* com conversor CC-CA com topologia meia-ponte.

Nos resultados, foi mostrado que se calcula de maneira muito simplificada os ganhos para o controlador ressonante, levando basicamente o tempo necessário para se editar as caixas de texto na interface. Foi mostrado também que, para um único modo ressonante aplicado ao controlador, a THD da saída é baixa (3,5%) quando aplicada carga resistiva. Como esperado, essa THD aumenta (11,6%) quando aplicada carga com corrente não linear. A adição de um segundo modo ressonante não exerce grande diferença na THD, que aproximadamente se mantém (3,3%) para carga linear e também para a carga não linear (11,3%). Porém, analisando a participação de cada componente na distorção, se observa grande redução componente na terceira harmônica (mesma frequência na qual foi adicionado novo compensador ressonante).

Assim, o uso da interface gráfica aqui proposta, atinge o objetivo de ser simples se comparado com o uso diretamente do *script* para solução do problema. Os ganhos obtidos através da interface, se aplicados ao controlador físico para ensaios, levam a *UPS* a imprimir na saída valores esperados de tensão e THD – comprovando que a comunicação entre o script e a interface é funcional, bem como comprovando que o primeiro realiza adequadamente o cálculo dos ganhos.

## 5.1. TRABALHOS FUTUROS

Ao desenvolver o trabalho, ficou claro que alguns pontos podem ser desenvolvidos a fim de obter melhores resultados. Os principais seriam:

- Tornar a interface gráfica mais geral: outros controladores aplicáveis à *UPS*, como o repetitivo [17], [18] e o repetitivo-ressonante [19], cujos cálculos dos parâmetros também são realizados por *scripts* em MatLab®, podem ser adicionados à interface gráfica. O trabalho é bastante semelhante ao apresentado aqui, onde haverá comunicação entre o script e uma interface gráfica. Uma janela de abertura mais completa que a aqui apresentada poderia permitir ao usuário qual controlador usar.

- Construir um modelo para simulação com o uso do Simulink®: a simulação utilizando o modelo no PSIM® é dificultada pela necessidade de estar rodando no computador ambos, MatLab® e PSIM®, de forma combinada. Diferentes versões destes programas apresentam diferentes incompatibilidades, e nem sempre é possível executar a simulação após o cálculo dos ganhos do controlador ressonante. Ainda, a simulação com o modelo aqui utilizado é bastante “lenta” se comparada às rotinas executadas dentro do ambiente MatLab® apenas, indicando que possivelmente, um modelo apenas no Simulink® seria não só muito mais simples de usar, mas executaria a simulação mais rapidamente.

- Adicionar bloco que permita ao usuário da interface, saber qual a situação atual do programa: indicar se está realizando captura de dados, simulação, entre outras informações.

- Caso se aplique nos blocos de entrada de dados, valores que não permitem solução para o controlador, o programa deve indicar o problema.

- Adicionar à interface gráfica a possibilidade de editar os valores do barramento DC (tensão de baterias da UPS) e da restrição em ângulo para as LMIs. Sobre o barramento DC, existe diretamente no *script* para cálculo dos ganhos esse dado, e bastaria mudar o mesmo para uma variável editável pela GUI. A restrição angular das LMIs iria requerer modificação no *script*, que apresenta tal parâmetro fixado em 90°.

- Reorganizar a GUI, colocando os dados referentes à simulação como parâmetros adicionados após o usuário confirmar que deseja ver a mesma (os degraus de carga não interferem na solução do problema, e podem ser solicitados caso o usuário execute a simulação). Ainda, é possível colocar quais dados se deseja visualizar graficamente, e não mantendo fixo apenas forma de onda e tensão RMS.

## REFERÊNCIAS

- [1] – **IEC 62040-3**: Uninterruptible Power Systems (UPS) - Part 3: Method of specifying the performance and test requirements. Switzerland: IEC, 2011. 1-214p.
- [2] - MANO, Otavio Simões. **Projeto de Controladores Ressonantes Para Sistemas Ininterruptos de Energia Para Seguimento Assintótico de Sinais de Referência Senoidais e Rejeição de Harmônicos**; 2010. Dissertação (graduação) – Pontifícia Universidade Católica do Rio Grande do Sul. Faculdade de Engenharia. Porto Alegre, 2010.
- [3] - LORENZINI, Charles. **Desenvolvimento de um Controlador Ressonante-Repetitivo Aplicado a Fontes Ininterruptas de Energia (UPS)**. 2015. Dissertação (Mestrado) – Universidade Federal do Rio Grande do Sul. Faculdade de Engenharia. Porto Alegre, 2015.
- [4] - ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Conversor a semicondutor - Sistema de alimentação de potência ininterrupta, com saída em corrente alternada (nobreak) - Terminologia**. ABNT. Rio de Janeiro. 2003. (NBR 15014).
- [5] - LI, P. Y. Internal Model Principle and Repetitive Control. In: **Advanced Control System Design**. [S.l.]: University of Minnesota, 2012.
- [6] - RECH, C. et al. Comparison of digital control techniques with repetitive integral action for low cost PWM inverters. **IEEE Transactions on Power Electronics**, Brookline, v.18, n.1, p.401–410, Jan 2003.
- [7] - WILLMANN, G. et al. Multiple-Loop H-Infinity Control Design for Uninterruptible Power Supplies. **IEEE Transactions on Industrial Electronics**, Brookline, USA, v.54, n.3, p.1–12, 2007.
- [8] - FUKUDA, S.; YODA, T. A novel current-tracking method for active filters based on a sinusoidal internal model [for PWM inverters]. **IEEE Transactions on Industry Applications**, Brookline, v.37, n.3, p.888–895, May 2001.
- [9] - FRANCIS, B.; WONHAM, W. The internal model principle for linear multivariable regulators. **Applied Mathematics and Optimization**, Berlin, v.2, n.2, p.170–194, 1975.

- [10] – PEREIRA, L. F. A. et al. Multiple Ressonant Controllers for Uninterruptible Power Supplies – A Systematic Robust Control Design Approach. **IEEE Transactions on Automatic Control**, Brookline, v.61, p. 1538, March 2014.
- [11] - THE MATHWORKS INC., **Creating Graphical User Interfaces**. Natick, USA. Março de 2015.
- [12] - MOSMANN, Bruno Henz. **Ensaio de Controladores de Alto Desempenho para Sistemas UPS**; 2014. Dissertação (graduação) – Universidade Federal do Rio Grande do Sul. Faculdade de Engenharia. Porto Alegre, 2014.
- [13] - IEC 61000-2-2: Electromagnetic compatibility (EMC) - Part 2-2: Environment - Compatibility levels for low-frequency conducted disturbances and signalling in public low-voltage power supply systems. Switzerland: IEC, 2002. 1-57p.
- [14] – mathworks.com. **Strings**. Outubro de 2015. Disponível em <<http://www.mathworks.com/help/matlab/ref/strings.html?searchHighlight=string>>. Acesso em: outubro de 2015
- [15] - ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **Conversor a semicondutor - Sistema de alimentação de potência ininterrupta com saída em corrente alternada (nobreak) - Segurança e desempenho**. ABNT. Rio de Janeiro. 2005. (NBR 15204).
- [16] - INSTITUTE OF ELECTRICAL AND ELECTRONIC ENGINEERS. **IEEE Recommended Practice for the application and testing of UPS for Power Generating Stations**. IEEE. New York, USA. 1986. (IEEE Std 944).
- [17] - FLORES, J. et al. Síntese de Controladores Repetitivos Chaveados: uma aplicação à fontes ininterruptas de energia (UPS). *Revista Controle & Automação*, Campinas, v.22, n.2, p.184–200, 2011.
- [18] - ESCOBAR, G. et al. Repetitive-Based Controller for a UPS Inverter to Compensate Unbalance and Harmonic Distortion. *IEEE Transactions on Industrial Electronics*, Brookline, v.54, n.1, p.504–510, Feb 2007.

[19] - ALVES PEREIRA, L. F. et al. A Resonant-Repetitive Control Scheme Applied to Uninterruptible Power Supplies (UPS). **Journal of Control, Automation and Electrical Systems**, Campinas, v. 24, p. 253-262, 2013.

## ANEXO – PRINCIPAIS “LINHAS DE PROGRAMAÇÃO” NO *GUI-GUIDE*

A maior parte da programação das *GUI* se dá de forma semelhante à programação dos *scripts*, onde os laços de repetição são baseados na linguagem C. Porém, conforme vemos em [referencia das *GUI*], capturar e escrever dados nos blocos da *GUI* tem um procedimento específico para cada caso. As linhas que foram mais frequentemente utilizadas no programa proposto nesse trabalho estão apresentadas abaixo:

### **Escrever dado em caixa de texto**

```
set(handles.textbox, 'String', variável);
```

Onde o termo “variável” possui um dado qualquer, que se deseja transferir para a caixa de texto com a TAG “textbox”.

### **Capturar dado de caixa de texto**

```
variavel = get(handles.textbox, 'String');
```

Onde o termo “variável” se refere à uma variável que irá receber o dado escrito na caixa de texto com a TAG “textbox”.

### **Capturar dado de grupo de botões mutuamente exclusivos**

```
variavel1 = get(handles.group_name, 'SelectedObject')
```

```
variavel2 = get(variavel1, 'String')
```

Onde o termo “variavel1” se refere a uma variável que irá receber o parâmetro ‘SelectedObject’ do grupo de botões com a TAG “group\_name”, e o termo “variavel2” se refere à uma variável que irá receber o dado no formato “String” salvo dentro da variável “variavel1”.

### **CheckBoxes**

```
variavel1 = get(handles.box1, 'Value')
```

```
variavel2 = get(handles.box2, 'Value')
```

Onde o termo “variavel1” se refere a uma variável que irá receber o parâmetro “value” escrito na checkbox com a TAG “box1”, e o termo “variavel2” se refere a uma variável que irá receber o parâmetro “value” escrito na checkbox com a TAG “box2”. O parâmetro “value” de uma checkbox será sempre um (caso a mesma esteja preenchida) ou zero (caso a mesma não esteja preenchida).

### **Desabilitar caixas CheckBoxes**

```
set(handles.box1, 'Enable', 'Off')
```

```
set(handles.box2, 'Enable', 'On')
```

Onde a checkbox com a TAG “box1” tem o parâmetro “Enable” definido como “off” (nesse caso, o usuário da GUI não poderá alterar o estado dessa checkbox), e a checkbox com a TAG “box2” tem o parâmetro “Enable” definido como “on” (nesse caso, o usuário da GUI passa a poder alterar o estado dessa checkbox).

### **Exportar variável**

```
assignin('base', 'variavel1', variavel_workspace)
```

Onde a GUI irá criar no *workspace* do MatLab® uma variável com o nome “variavel\_workspace”, e irá escrever os dados presentes na variável “variavel1” na primeira.

### **Importar variável**

```
variavel1 = EVALIN('base', 'variavel2')
```

A GUI irá criar uma variável dentro de seu próprio ambiente, com o nome “variavel1”, e nela, irá escrever os dados presentes na variável chamada “variavel2”, que deve estar no *workspace* do MatLab®. Caso não exista a variável “variavel2” no *workspace*, quando a GUI executar esse comando ela irá apresentar mal funcionamento.

### **Plot de gráfico**

```
axes(handles.axes1)
```

```
plot(a, b)
```

A GUI irá imprimir em seu bloco ‘axes’ com a TAG “axes1” o gráfico definido pelo comando ‘plot’. Assim, será apresentado na saída o gráfico do vetor ‘a’ pelo dado do vetor ‘b’. Linhas buscando organizar a área do gráfico, contendo comandos como ‘xgrid’, ‘xlabel’, ‘ylabel’, etc, podem ser adicionadas abaixo da linha com o comando ‘plot’.