

**MARCUS VINÍCIUS RICHARDELLE UNZUETA**

**UMA CONTRIBUIÇÃO AO ESTUDO DAS REDES MUTUAMENTE  
CONECTADAS DE DPLLS USANDO MODELOS DE TEMPO  
DISCRETO**

**Texto apresentado à Escola  
Politécnica da Universidade de São  
Paulo para obtenção do título de  
Mestre em Engenharia.**

**Área de Concentração:  
Engenharia Elétrica**

**Orientador:  
Prof. Dr. José Roberto Castilho  
Piqueira**

**São Paulo - SP  
2008**

# **Agradecimentos**

Ao professor José Roberto Castilho Piqueira pelo apoio.

Aos meus pais, irmãos e amigos.

# Resumo

Este trabalho tem por objetivo apresentar uma nova forma de analisar as redes de sincronismo de fase mutuamente conectadas. Estas redes são formadas por “Phase-Locked Loops” digitais – ou DPLLs. O sinal gerado por cada DPLL é enviado a todos os demais dispositivos, formando a rede mutuamente conectada.

Parte-se do pressuposto de que as ligações entre os dispositivos são dotadas de atrasos, o que dificulta o tratamento do problema. No entanto, é apresentado aqui um método para análise das malhas de sincronismo via discretização do modelo de tempo contínuo, objetivando dirimir essa dificuldade, já que atrasos são facilmente representados em modelos de tempo discreto.

Para tanto, o modelo da rede no espaço de estados é equacionado a partir da rede. Esse modelo no espaço de estados é, então, discretizado e, enfim, pode-se determinar o estado síncrono da rede – incluindo a frequência de sincronismo – e analisar sua estabilidade.

Como se poderá constatar, escolhendo um período de amostragem adequado, pode-se representar o comportamento das redes de sincronismo com modelos discretos, obtendo elevado grau de precisão.

# Abstract

This work introduces a new method for studying a mutually-delayed-connected network of Digital Phase-Locked Loops – DPLLs. The signal generated by a DPLL in the network is sent to all other devices in this same network.

Because of delayed signals, it is difficult to treat this problem. So, it's shown here a method for analyzing the networks via discretization of continuous time delay model in order to deal with this issue easily, considering that delays are naturally represented in discrete time models.

First of all, a continuous state space model is obtained from mutually-connected network. Then, this model is discretized and, finally, the synchronous state can be determined and the stability can be analyzed.

As shown below, choosing a proper time sample, the behavior of mutually-delayed-connected networks can be approximately represented by a discrete time model.

# Lista de Figuras

Figura 1-1: Exemplo de uma topologia de rede.....	10
Figura 2-1: Diagrama em blocos de um PLL. ....	12
Figura 2-2: Forma de onda gerada pela porta OU-EXCLUSIVO. ....	14
Figura 2-3: a) Bloco lógico; b) Exemplo de resposta aos sinais de entrada. ....	15
Figura 2-4: Duty cycle de saída do detector de fases flip-flop JK.....	17
Figura 2-5: Largura de pulso na saída do detector de fases “charge-pump”.....	19
Figura 2-6: Resposta do detector “charge-pump”: a) $0 < \theta_e < 2\pi$ ; b) $-2\pi < \theta_e < 0$ . ....	20
Figura 2-7: Resposta do comparador para o caso onde $\omega_{in} > \omega_{out}$ . ....	21
Figura 2-8: Filtro de primeira ordem formado por resistor e capacitor. ....	23
Figura 2-9: Filtro de primeira ordem modificado para adequação ao detector “charge-pump”. .....	24
Figura 2-10: Modelo linear em blocos do DPLL.....	25
Figura 3-1: Topologia de uma rede com três nós mutuamente sincronizados e totalmente conectados.....	28
Figura 3-2: DPLL com duas entradas ponderadas.....	29
Figura 3-3: DPLL com N-1 entradas ponderadas. ....	30
Figura 3-4: Modelo do DPLL com duas entradas ponderadas. ....	31
Figura 4-1: Entrada $u(t)$ atrasada. ....	42
Figura 4-2: Aproximação do estado de $x(t)$ .....	44
Figura 5-1: Rede com quatro nós mutuamente conectados.....	52
Figura 5-2: Malha com quatro nós: resposta do modelo de tempo contínuo do caso 1.....	54
Figura 5-3: Malha com quatro nós: resposta do modelo de tempo discreto do caso 1.....	56
Figura 5-4: Malha com quatro nós: resposta do erro contínuo $\times$ discreto do caso 1.....	57
Figura 5-5: Malha com quatro nós: resposta do sistema real do caso 1.....	57
Figura 5-6: Malha com quatro nós: resposta real $\times$ discreto do caso 1.....	58

Figura 5-7: Malha com quatro nós: resposta do modelo de tempo contínuo do caso 2.....	61
Figura 5-8: Malha com quatro nós: resposta do modelo de tempo discreto do caso 2.....	63
Figura 5-9: Malha com quatro nós: resposta do erro contínuo × discreto do caso 2.....	63
Figura 5-10: Malha com quatro nós: resposta do sistema real do caso 2.....	64
Figura 5-11: Malha com quatro nós: resposta real × discreto do caso 2.....	65
Figura 5-12: Rede com cinco nós mutuamente conectados.....	65
Figura 5-13: Malha com cinco nós: resposta do modelo de tempo contínuo do caso 1.....	68
Figura 5-14: Malha com cinco nós: resposta do modelo de tempo discreto do caso 1.....	70
Figura 5-15: Malha com cinco nós: resposta do erro contínuo × discreto do caso 1.....	71
Figura 5-16: Malha com cinco nós: resposta do sistema real do caso 1.....	71
Figura 5-17: Malha com cinco nós: resposta do modelo real × discreto do caso 1.....	72
Figura 5-18: Malha com cinco nós: resposta do modelo de tempo contínuo do caso 2.....	75
Figura 5-19: Malha com cinco nós: resposta do modelo de tempo discreto do caso 2.....	76
Figura 5-20: Malha com cinco nós: resposta do erro contínuo × discreto do caso 2.....	77
Figura 5-21: Malha com cinco nós: resposta do sistema real do caso 2.....	77
Figura 5-22: Malha com cinco nós: resposta real × discreto do caso 2.....	78

# Sumário

<b>1 Introdução .....</b>	<b>9</b>
1.1 Metodologia do trabalho .....	11
1.2 Estrutura do Trabalho.....	11
<b>2 O Modelo do DPLL .....</b>	<b>12</b>
2.1 O Detector de Fase.....	13
2.1.1 Porta Lógica OU Exclusivo .....	14
2.1.2 Flip-flop JK.....	15
2.1.3 Detector “Charge-Pump”.....	17
2.2 O Filtro .....	22
2.3 O Oscilador Controlado por Tensão .....	24
2.4 O Modelo Completo do DPLL .....	25
2.4.1 Sistema Reduzido.....	26
<b>3 As Redes Mutuamente Conectadas.....</b>	<b>28</b>
3.1 Ponderação de Fase.....	29
3.2 Redes com Três Nós .....	30
3.2.1 Modelo no Domínio do Tempo.....	30
3.3 Redes com N nós .....	32
3.3.1 Estado Síncrono .....	32
3.4 Representação no Espaço de Estados.....	34
3.4.1 Representação de uma Rede de Três Nós.....	35
<b>4 Discretização.....</b>	<b>39</b>
4.1 A Entrada.....	41
4.1.1 Entrada sem Atrasos.....	41
4.1.2 Entrada com Atrasos .....	42
4.2 A Realimentação de Estados .....	43
4.2.1 De 0 a $c_i$ .....	44
4.2.2 De $c_i$ a $T_s$ .....	45
4.2.3 Determinação dos termos $^{pos}A_{di}$ , $^{act}A_{di}$ , e $^{pre}A_{di}$ .....	46
4.3 A Saída .....	47
4.4 Modelo de Tempo Discreto e Regime Permanente .....	47

4.5	Modelo de Tempo Discreto e Estabilidade .....	50
<b>5</b>	<b>Resultados e Simulações.....</b>	<b>51</b>
5.1	Redes com Quatro Nós.....	52
5.1.1	Caso 1 .....	52
5.1.2	Caso 2 .....	59
5.2	Redes com Cinco Nós .....	65
5.2.1	Caso 1 .....	66
5.2.2	Caso 2 .....	73
<b>6</b>	<b>Conclusão.....</b>	<b>79</b>
	<b>Referências Bibliográficas.....</b>	<b>81</b>
	<b>A Fórmulas de Integração.....</b>	<b>83</b>
	<b>B Listagens .....</b>	<b>86</b>



# Capítulo 1

## Introdução

As malhas ou redes de sincronismo são constituídas por diversos relógios ou osciladores espalhados geograficamente. Esses relógios conectam-se de acordo com diferentes topologias de modo a atingir um estado síncrono, em que as frequências de oscilação são idênticas.

Denominando cada relógio por *nó*, o fato de os nós estarem separados sobre uma determinada região introduz atrasos de transmissão dos sinais gerados, devidos à distância entre um relógio e outro.

Dentre as aplicações das malhas de sincronismo, pode-se citar:

- distribuição de sinais de tempo para processadores;
- sincronização de relógios para redes de comunicação;
- geração de padrões de tempo para controle de processos.

As razões apresentadas são suficientes para justificar o estudo dessas redes. De fato, a estabilidade dos pontos de equilíbrio e a determinação desses pontos já foram tratados em trabalhos anteriores a este ([1] a [5]).

A abordagem que será adotada neste trabalho para as malhas de sincronismo é a representação da dinâmica dos nós, bem como suas interligações, através de modelos de tempo discreto. Esse tipo de representação já é utilizado na prática ([9] e [10]) devido à disseminação em grande escala dos sistemas de controle e de tratamento de sinais digitais.

Outra motivação para se fazer a ponte entre as abordagens de tempo contínuo e discreto no processamento de sinais ([9]) é a existência de atrasos inerentes à transmissão entre dois pontos distintos pertencentes ao mesmo circuito.

O fato de os sistemas de tempo discreto serem baseados principalmente em tempos de atraso motivou sua utilização no tratamento do problema das malhas de sincronismo de fase mutuamente conectadas. A discretização desses sistemas, intrinsicamente contínuos, pode ser de grande utilidade na análise da estabilidade em torno dos pontos de equilíbrio, similarmente à análise já feita para as redes sem o artifício da discretização.

A Figura 1-1 mostra um exemplo de rede formada por nós, sendo cada nó um PLL – Phase-Locked Loop. Vale frisar que a distribuição geográfica, que insere atrasos de transmissão entre dois nós quaisquer, representa uma dificuldade a mais no estudo do problema no tempo contínuo, além daquelas devidas às não-linearidades.

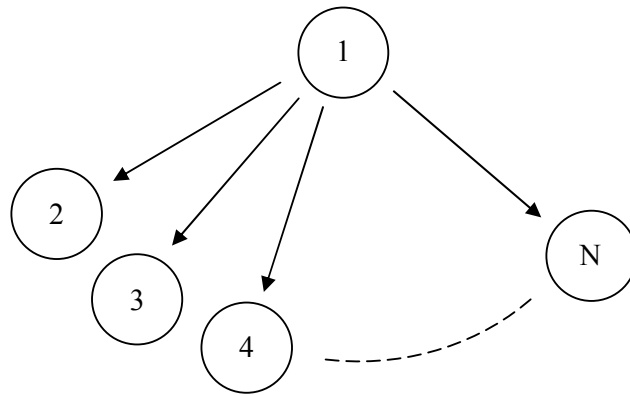


Figura 1-1: Exemplo de uma topologia de rede.

Em ([16]), está descrito um método de obtenção de um modelo discreto, a partir de um sistema contínuo invariante no tempo. Tal método leva em conta a já conhecida solução do sistema contínuo, dada por um equação que determina exatamente o valor do estado em função do vetor de entradas e do tempo. Essa é a forma mais simples de discretização, por não levar em conta quaisquer atrasos.

Neste trabalho, pretende-se representar as malhas de sincronismo por um modelo no espaço de estados, primeiramente contínuo e, em seguida, seu equivalente de tempo discreto. No modelo de tempo contínuo serão levados em consideração os tempos de propagação. Com a discretização desse modelo, espera-se dirimir as dificuldades introduzidas pelos atrasos – também conhecidos como *tempos mortos* – para que a dinâmica do sistema possa ser tratada de maneira simplificada.

## 1.1 Metodologia do trabalho

Como a proposta deste trabalho é utilizar uma nova abordagem no estudo da estabilidade das redes mutuamente conectadas em torno do estado síncrono, tratando o problema de forma discretizada, o primeiro passo é estabelecer o modelo contínuo do DPLL, que será utilizado nas malhas de sincronismo. Como visto adiante, serão utilizados dois tipos diferentes de detectores de fase e filtros de primeira ordem.

A seguir será definida a topologia da malha a ser estudada, constituída de três a cinco nós totalmente conectados. Para dar continuidade, dever-se-á obter um modelo no espaço de estados.

Será proposta uma forma de discretização do sistema contínuo no espaço de estados dotado de entradas e saídas atrasadas – atrasos externos – e realimentado por estados atrasados no tempo – atrasos internos. O método de discretização será construído de forma genérica, podendo ser aplicado a qualquer sistema contínuo linear e invariante no tempo (LTI). Isso quer dizer que o método de discretização é aplicável a qualquer modelo LTI contínuo, que utiliza ou não atrasos, sejam eles internos ou externos.

Por fim, serão apresentados exemplos de redes, com resultados de simulação de modelos reais, contínuos e discretos.

## 1.2 Estrutura do Trabalho

Este trabalho está dividido da seguinte forma:

- Capítulo 2: É obtido o modelo do DPLL, que será utilizado em cada um dos nós da rede de sincronismo.
- Capítulo 3: São apresentadas as topologias de rede a serem analisadas.
- Capítulo 4: É apresentado o método de discretização do sistema contínuo dotado de atrasos nas entradas e estados.
- Capítulo 5: São apresentados alguns resultados via simulação computacional.
- Capítulo 6: Apresentação resumida das contribuições deste trabalho.

## Capítulo 2

### O Modelo do DPLL<sup>1</sup>

As redes de sincronismo são baseadas em dispositivos conhecidos como “Phase-Locked Loops”, ou PLL. Portanto, convém abordar seus princípios de funcionamento para depois considerá-lo na descrição de funcionamento das redes.

Os PLL são muito utilizados na detecção de bases de tempo. Entre algumas de suas aplicações, pode-se citar modulação e demodulação FM, decodificação de tom, filtro de frequências entre outras.

Basicamente, o PLL é um circuito capaz de gerar um sinal de tempo sincronizado em fase e frequência<sup>1</sup> com um sinal de referência externo.

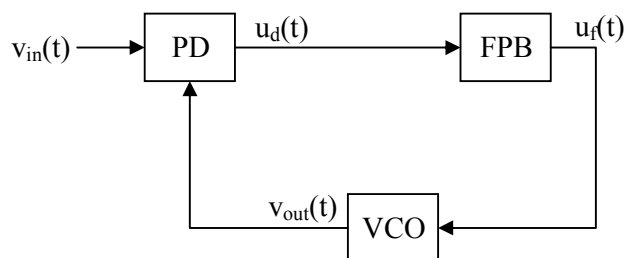


Figura 2-1: Diagrama em blocos de um PLL.

O PLL é dotado de um detector de fase (PD), um filtro passa-baixas (FPB) e um oscilador controlado por tensão (VCO). O PD é responsável por medir a diferença de fase entre o sinal de entrada  $v_{in}(t)$  e o sinal gerado pelo VCO  $v_{out}(t)$ , fornecendo uma saída apropriada. O FPB

<sup>1</sup> ([18])

elimina as componentes de alta frequência geradas pelo PD. O sinal de saída do filtro  $u_f(t)$  controla a frequência do VCO.

Um PLL encontra-se em estado síncrono quando atinge um ponto de equilíbrio, de modo que o erro de frequência seja nulo. Supondo o sistema em estado síncrono estável, uma diferença de fases entre as formas de onda tratadas pelo PD – provocada por uma variação na frequência do sinal de entrada – altera a saída do detector de fase. A tensão de saída do filtro irá variar a frequência do VCO de modo a corrigir a diferença de frequências e, conseqüentemente, a diferença de fases.

Os osciladores possuem uma faixa de trabalho constituída de uma frequência máxima e uma mínima. O ponto médio desta faixa é denominado *frequência central* ou *frequência natural*.

Os PLLs operavam originalmente com sinais analógicos, formados por ondas senoidais. O DPLL – “Digital Phase Locked Loop” – é um tipo especial de PLL cuja operação está baseada em formas de onda digitais ou quadradas. Dentre as qualidades de um PLL digital, pode-se citar:

- melhor imunidade a ruído;
- facilidade de aquisição;
- baixo custo.

A seguir será detalhado o funcionamento de cada um dos blocos do DPLL, obtendo-se gradativamente seu modelo linear aproximado. A partir deste modelo linear, será construído o modelo da malha de sincronismo para posterior discretização.

## 2.1 O Detector de Fase<sup>2</sup>

Os detectores de fase digitais são divididos em três (16):

- a) porta lógica OU Exclusivo;
- b) flip-flop JK;
- c) detector “Charge-Pump”.

Os detectores (a) e (b) serão mais brevemente apresentados pois, a princípio, não serão utilizados na composição das redes, tema do próximo capítulo. O detector (c) será mostrado com maiores detalhes, pois possui uma faixa maior de operação, como visto adiante, o que pode ser vantajoso na composição das malhas de sincronismo.

---

<sup>2</sup> ([18])

### 2.1.1 Porta Lógica OU Exclusivo<sup>3</sup>

Esse circuito lógico possui a seguinte tabela da verdade:

Tabela 2-1		
$v_i$	$v_o$	$u_d$
0	0	0
0	1	1
1	0	1
1	1	0

Considerando dois sinais distintos  $v_{in}$  e  $v_{out}$  como entradas desse tipo de porta, quando os níveis lógicos de ambos forem diferentes, o nível lógico na saída é 1. Quando os sinais de entrada possuírem o mesmo nível lógico, a saída  $u_d$  vai a 0. Na Figura 2-2 está um exemplo do comportamento deste tipo de porta lógica.  $\theta_e$  representa o erro de fase entre os dois sinais de entrada.

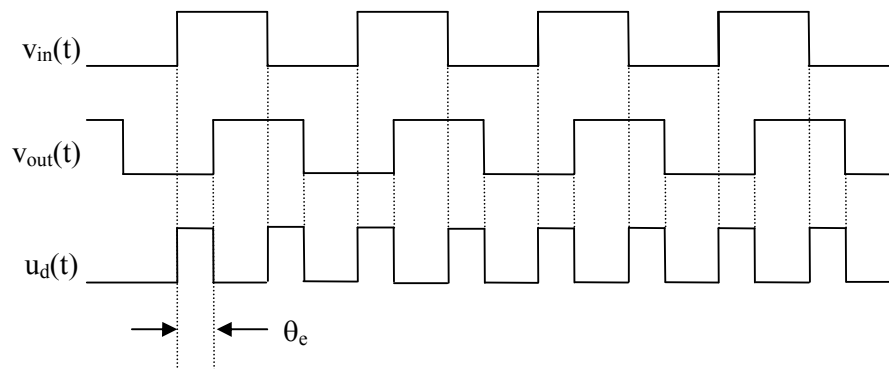


Figura 2-2: Forma de onda gerada pela porta OU-EXCLUSIVO.

Na frequência central, esse comparador mantém uma diferença de fase de  $90^\circ$  entre o sinal de entrada e o sinal do VCO – que representa o erro de fase nulo –, considerando o “duty cycle” igual a 50% para os sinais tratados. Nessa situação de regime dizemos que o DPLL está em estado de “lock”. A saída linear aproximada desse detector é dada por:

$$u_d = \frac{V_{cc}}{\pi} \theta_e = \frac{V_{cc}}{\pi} (\theta_{in} - \theta_{out}), \quad (2-1)$$

sendo  $\theta_e$  o erro de fase e considerando que o nível dos sinais variam entre 0 e  $V_{cc}$ . Vale notar que os sinais eletrônicos propriamente ditos –  $v_{in}$  e  $v_{out}$  – não aparecem na equação, apenas suas fases.

<sup>3</sup> ([18] e [26])

A faixa de operação desse detector é a menor dentro os três citados, e vai de  $-\pi/2$  a  $\pi/2$ . Por esse fato, preferiu-se não utilizá-lo para compor os DPLLs.

### 2.1.2 Flip-flop JK<sup>4</sup>

Trata-se de um dispositivo que detecta a transição positiva ou negativa dos sinais de entrada, ou seja, quando aplicamos uma borda de subida na entrada J desse “flip-flop”, sua saída Q vai a nível lógico 1 – caso ainda não esteja – e quando aplicamos uma borda de descida na entrada K, a saída Q vai a nível 0. A Figura 2-3b mostra um exemplo de resposta para sinais de entrada adequados em J e K.

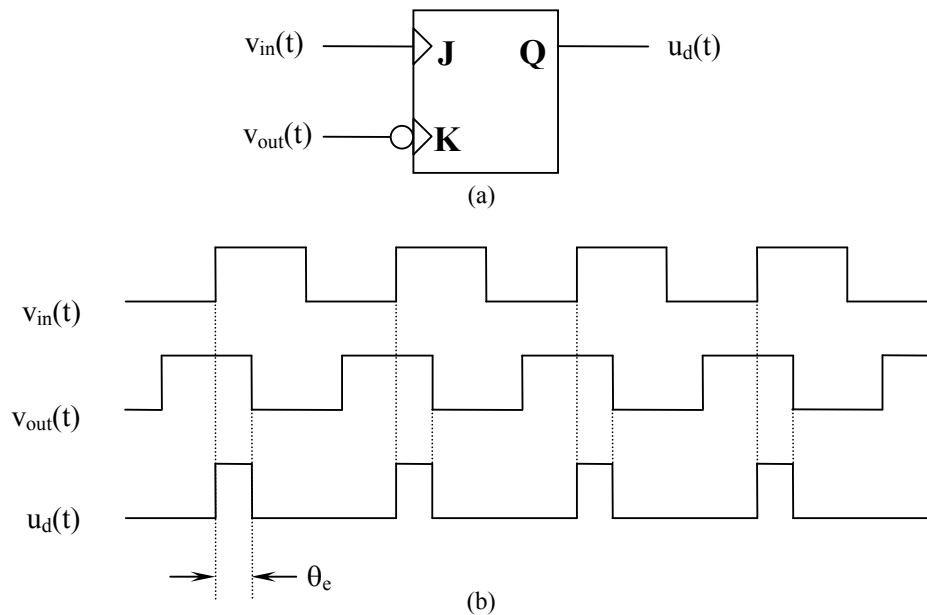


Figura 2-3: a) Bloco lógico; b) Exemplo de resposta aos sinais de entrada.

A Tabela 2-2 mostra o diagrama de entradas e saídas do flip-flop. O símbolo  $\uparrow$  representa uma borda de subida,  $\downarrow$  representa uma borda de descida e X representa um estado qualquer.

$u_{d\ k-1}$	J	K	$u_{d\ k}$
0	$\uparrow$	X	1
0	X	$\downarrow$	0
1	$\uparrow$	X	1
1	X	$\downarrow$	0

<sup>4</sup> ([18])

$u_{d\ k-1}$  e  $u_{d\ k}$  representam, respectivamente, a saída anterior e a saída atual do dispositivo.

O sinal de interesse, na saída deste detector, é representado pela largura de pulso da forma de onda gerada – o “duty cycle”. Sabendo-se que:

$$\theta_e = \theta_{in} - \theta_{out}, \quad (2-2)$$

temos o “duty cycle” DC definido como:

$$DC = 50\% + \frac{\theta_e}{2\pi}. \quad (2-3)$$

Pela equação (2-3), e considerando que  $\theta_e$  varia de  $-\pi$  a  $+\pi$ , nota-se que o “duty cycle” varia de 0% a 100%. O nível de tensão médio da saída depende dos níveis máximo  $V_{max}$  e mínimo  $V_{min}$  da tensão de saída, resultando:

$$u_d = (V_{max} - V_{min}) \left( \frac{\pi + \theta_e}{2\pi} \right) + V_{min} \quad (2-4)$$

Normalizando a resposta do dispositivo em torno de 50%, e fazendo  $V_{max} = -V_{min} = V$  a saída  $u_d$  do comparador de fases, em função da defasagem entre os sinais de entrada  $\theta_e$ , fica:

$$u_d = K_d \theta_e, \quad (2-5)$$

em que

$$K_d = \frac{V}{\pi}. \quad (2-6)$$

A saída do comparador de fases é aplicada a um filtro que irá atenuar as componentes de alta frequência, fornecendo um sinal cujo valor médio é representado por  $u_d$ .

Estando o PLL sincronizado, se a diferença de fases é nula – com o “duty cycle” a 50%, ponto sobre o qual normalizamos a saída do detector – o PLL está sintonizado em sua frequência central, que representa o ponto médio dos possíveis valores que o VCO pode gerar. O detalhamento do VCO é feito no item 2.3.

Imaginando a hipótese de se ter uma defasagem de sinal  $|\theta_e| > \pi$ , observa-se um comportamento fortemente não-linear do detector. A resposta do duty cycle encontra-se na figura a seguir.



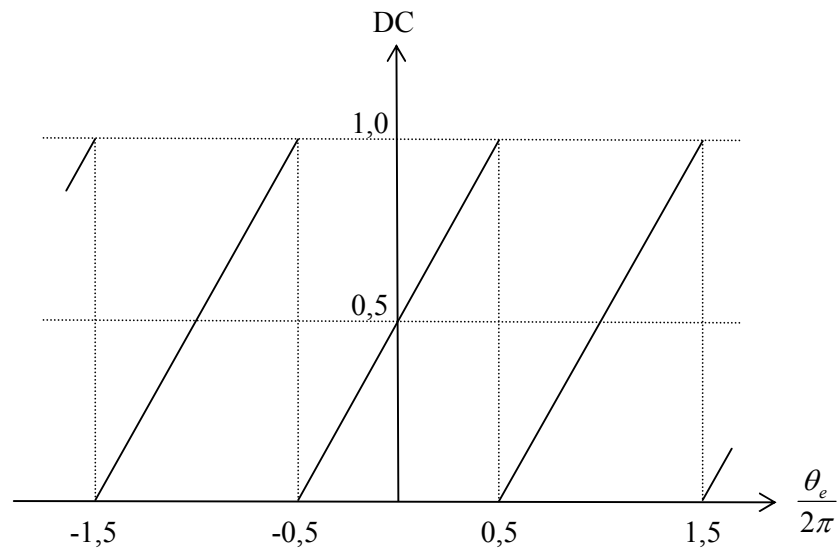


Figura 2-4: Duty cycle de saída do detector de fases flip-flop JK.

Como se pode notar, esse detector possui uma faixa de operação que vai de  $-\pi$  a  $+\pi$ , substancialmente maior que o OU-Exclusivo. Porém, cabe ressaltar que ambos possuem um comportamento aproximadamente linear, apesar das grandes diferenças nas faixas de operação.

### 2.1.3 Detector “Charge-Pump”<sup>5</sup>

Este comparador, assim como o “flip-flop” JK, efetua a detecção de fase usando as transições de borda dos sinais de entrada. Porém ambas as entradas são ativadas por borda de subida de sinal. Sua saída  $u_d$  apresenta, além dos níveis lógicos 0 e 1, um terceiro estado conhecido como “tri-state”. Esse último caracteriza-se pela alta-impedância no terminal do dispositivo.

O nome “charge-pump” deve-se à sua utilização estar associada a um circuito integrador, ativo ou passivo. Quando a saída  $u_d$  é igual a  $V_{max}$  ou  $V_{min}$ , o sinal na saída do integrador é uma rampa em aclave ou declive, respectivamente. Quando  $u_d$  encontra-se no estado de alta impedância, não há circulação de corrente e o integrador mantém seu nível ou *carga*. Quanto maior ou menor o valor de  $\theta_e$ , maior ou menor é a taxa de variação na saída do integrador. Essas características fazem com que esse de comparador seja capaz de efetuar o sincronismo do DPLL tanto em fase quanto em frequência, ou seja, dentro de sua faixa de operação, o DPLL é conduzido a um estado síncrono com erro de fase nulo.

<sup>5</sup> ([18] e [26])

No entanto, esta situação é indesejável pois, como será visto mais adiante, uma rede em seu estado síncrono pressupõe erros de fase não nulos nos detectores. Nessa condição, um integrador pode facilmente saturar sua saída, em caso de uma implementação prática. Basta lembrar que em circuitos eletrônicos, um integrador – normalmente implementado com um capacitor – possui tensões máxima e mínima de acordo com a alimentação, a saber,  $V_{max}$  e  $V_{min}$ . Felizmente, essa situação pode ser contornada efetuando-se simples alterações, que serão mostradas no item 2.2 quando os filtros forem apresentados. Isto implica em erros de fase que podem não ser nulos no sincronismo em frequência – o que é desejável – e possibilita sua aplicação sem restrição da faixa de operação.

O estado da saída deste detector depende, além das entradas, do estado anterior e da própria saída. Dessa forma, pode-se construir a tabela de comportamento, como visto na Tabela 2-3. O símbolo  $\uparrow$  representa uma borda de subida e X representa um nível qualquer.  $v_{in}$  é o sinal de entrada e  $v_{out}$  é o sinal gerado pelo VCO.  $u_{d\ k-1}$  e  $u_{d\ k}$  representam, respectivamente, a saída anterior e a saída atual do dispositivo.

$u_{d\ k-1}$	$v_{in}$	$v_{out}$	$u_{d\ k}$
0	X	X	0
0	X	$\uparrow$	0
0	$\uparrow$	X	3-state
0	$\uparrow$	$\uparrow$	0
3-state	X	X	3-state
3-state	X	$\uparrow$	0
3-state	$\uparrow$	X	1
3-state	$\uparrow$	$\uparrow$	3-state
1	X	X	1
1	X	$\uparrow$	3-state
1	$\uparrow$	X	1
1	$\uparrow$	$\uparrow$	1

Existem duas situações sobre as quais deve-se analisar o funcionamento deste comparador. A primeira é quando as frequências de entrada e de saída,  $\omega_i$  e  $\omega_o$ , são iguais, quando o detector está próximo de seu estado síncrono. A segunda é quando as frequências de entrada e saída são diferentes. Abaixo encontra-se cada uma destas situações.

$$a) \quad \omega_{in} = \omega_{out} e |\theta_e| < 2\pi$$

Nesse caso, o regime caracteriza-se pela diferença de fase de  $0^\circ$ , entre a saída e a entrada, com a peculiaridade de esse estado ocorrer mesmo fora da frequência central. O comportamento do sistema é não-linear, como visto na Figura 2-5.

Porém, o detector de fase-frequência, como também é conhecido, tem um comportamento aproximadamente linear para uma faixa de operação que vai de  $-2\pi$  a  $+2\pi$ , diferentemente do “flip-flop”. Isto ocorre devido aos três estados possíveis que sua saída pode apresentar.

Quando  $0 < \theta_e < 2\pi$ , o sinal de saída varia entre  $V_{max}$  e o estado de alta impedância, com a largura de pulso indicada pela Figura 2-5. Quando  $-2\pi < \theta_e < 0$ , o sinal de saída varia entre  $V_{min}$  e o estado de alta impedância, com a largura de pulso proporcional a  $\theta_e$ . Vale notar que a figura possui dois eixos simétricos “UP” e “DOWN”, que podem ser pensados como dois duty cycles diferentes. A Figura 2-6 ilustra o comportamento do comparador nesses casos.

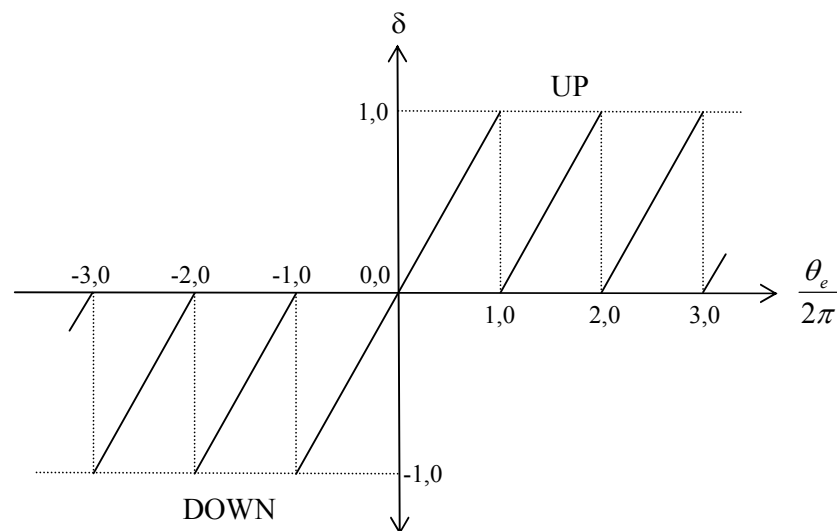


Figura 2-5: Largura de pulso na saída do detector de fases “charge-pump”.

A largura de pulso  $\delta$  – que substitui o “duty cycle” visto anteriormente – pode ser modelada da seguinte forma:

$$\delta = K_d \theta_e \quad (2-7)$$

sendo  $\theta_e = \theta_{in} - \theta_{out}$ . O valor de  $\delta$  pode variar de  $-1$  a  $+1$ . Quando  $0 < \delta < 1$ , a saída do comparador varia entre  $V_{max}$  e o estado de alta impedância, que compreende o semi-plano denominado “UP” na Figura 2-5. Com  $-1 < \delta < 0$ , a saída do comparador varia entre  $V_{min}$  e o

estado de alta impedância, que compreende o semi-plano denominado “DOWN” na Figura 2-5.

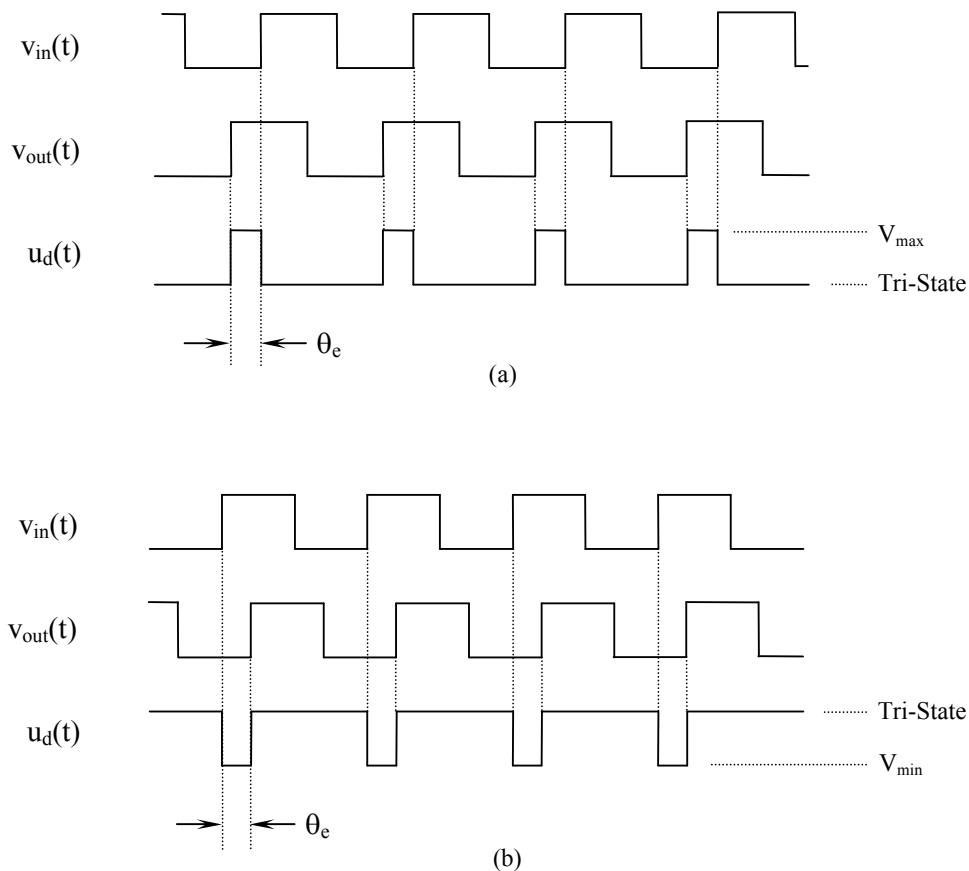


Figura 2-6: Resposta do detector “charge-pump”: a)  $0 < \theta_e < 2\pi$ ; b)  $-2\pi < \theta_e < 0$ .

Os fatos da largura de pulso ser uma função linear da diferença de fase dos sinais de entrada em torno do estado síncrono, e do circuito possuir uma ampla faixa de operação são determinantes na sua utilização nas malhas de sincronismo.

b)  $\omega_{in} \neq \omega_{out}$

Nessa situação, não há como estabelecer o valor da diferença de fases entre os sinais de entrada e saída. Conseqüentemente, a largura de pulso também não pode ser precisamente determinada.

A solução encontrada é determinar o valor médio de  $\delta$ , como feito em ([18]), que iremos denominar  $\delta_m = E[\delta(t)]$ .

Para tanto, vamos supor que a frequência  $\omega_{in}$  seja maior que  $\omega_{out}$ . No instante inicial, ocorre uma borda positiva no sinal de entrada, o que leva a saída a nível 1, implicando uma saída igual a  $V_{max}$  no comparador. O comparador voltará ao estado de alta impedância quando ocorrer uma borda positiva no sinal  $v_{out}$ . Porém, a partir do instante inicial até o instante igual ao período do sinal de saída  $T_{out}$ , a ocorrência de uma borda positiva em  $v_{out}$  possui uma densidade de probabilidade  $p_{out}$ . Esta distribuição é, supostamente, uniforme, como visto na Figura 2-7, e vale:

$$p_{out}(t) = \frac{1}{T_{out}} \quad (2-8)$$

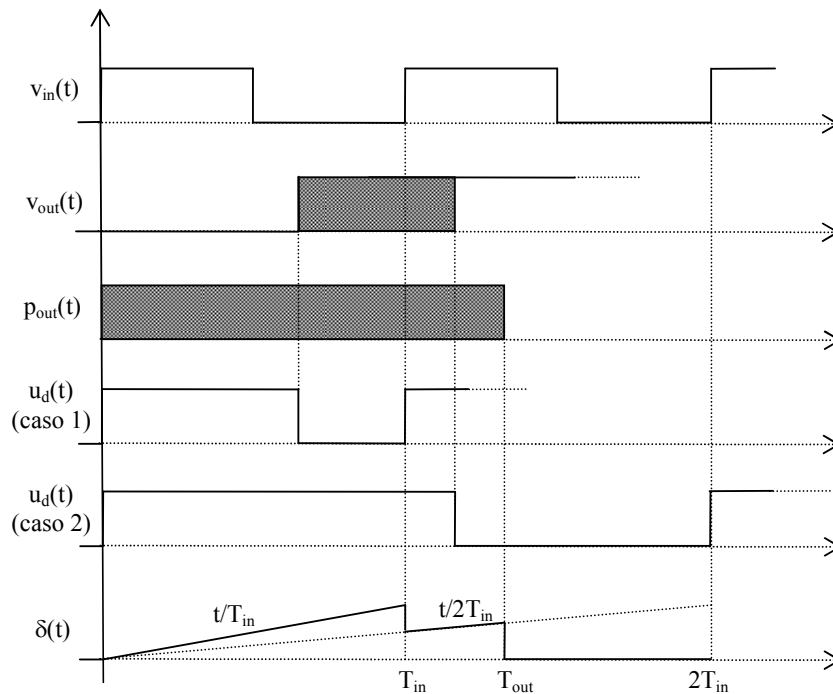


Figura 2-7: Resposta do comparador para o caso onde  $\omega_{in} > \omega_{out}$ .

Na mesma Figura 2-7, pode-se observar dois casos distintos para a saída:

- Caso 1: Transição positiva do sinal  $v_{out}(t)$  em  $0 \leq t \leq T_{in}$ ;
- Caso 2: Transição positiva do sinal  $v_{out}(t)$  em  $T_{in} \leq t \leq T_{out}$ .

No primeiro caso, a largura de pulso  $\delta$  é dada por:

$$\delta(t) = \frac{t}{T_{in}} ; 0 \leq t \leq T_{in}, \quad (2-9)$$

e no segundo caso, a largura de pulso  $\delta$  é dada por:

$$\delta(t) = \frac{t}{2T_{in}} ; T_{in} \leq t \leq T_{out}. \quad (2-10)$$

Neste exemplo, o período  $T_{out}$  é maior que  $T_{in}$ , porém  $T_{out} < 2T_{in}$ . O valor médio  $\delta_{med}$  é dado por:

$$E[\delta(t)] = \delta_{med} = \int_0^{T_{out}} p_{out}(t) \delta(t) dt. \quad (2-11)$$

A integral pode ser escrita como:

$$\delta_{med} = \int_0^{T_{in}} p_{out}(t) \frac{1}{T_{in}} dt + \int_{T_{in}}^{T_{out}} p_{out}(t) \frac{1}{2T_{in}} dt. \quad (2-12)$$

Para generalizar o que foi mostrado, deve-se pensar a possibilidade de que  $\omega_{in}$  seja maior que  $l\omega_{out}$ , com  $l = 1, 2, 3, 4, \dots$ . Assim, baseado na equação (2-12), tem-se:

$$\delta_{med} = \int_0^{T_{in}} p_{out}(t) \frac{1}{T_{in}} dt + \int_{T_{in}}^{2T_{in}} p_{out}(t) \frac{1}{2T_{in}} dt + \dots + \int_{(l-1)T_{in}}^{T_{out}} p_{out}(t) \frac{1}{lT_{in}} dt, \quad (2-13)$$

e, conforme (2-8), trabalhando a equação (2-13):

$$\delta_{med} = f_{out} \left( \sum_{j=1}^l \frac{1}{j} - 1 \right) + \frac{f_{in}}{l}, \quad (2-14)$$

sendo  $f_{in}$  e  $f_{out}$  as frequências de entrada e saída, respectivamente. Na equação (2-14),  $l = \text{int}(\omega_{out}/\omega_{in}) + 1$ , sendo  $\text{int}(\sigma)$  o menor inteiro maior que  $\sigma$ . O caso em que  $\omega_{out} > \omega_{in}$  pode ser obtido de modo similar.

Nota-se que, quando  $\omega_{in}$  é muito maior que  $\omega_{out}$  ou  $\omega_{in}$  é muito menor que  $\omega_{out}$ , o comportamento da largura de pulso é fortemente não-linear. Essas situações não são desejáveis em nossa aplicação, apesar de possíveis. Portanto, será considerado apenas o caso em que ambas as frequências de entrada e saída são aproximadamente iguais –  $\omega_{in} \cong \omega_{out}$ .

## 2.2 O Filtro

A principal função do filtro é eliminar os ruídos de alta frequência provenientes do detector de fase, resultando em sinal que melhor identifique a diferença de fases entre as formas de onda de entrada e do VCO.

A Figura 2-8 mostra um filtro de primeira ordem – que possui apenas um polo – formado por um resistor e um capacitor.

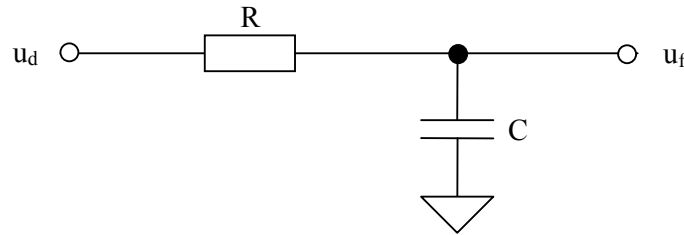


Figura 2-8: Filtro de primeira ordem formado por resistor e capacitor.

Na Figura 2-8,  $u_d$  é a saída do detector de fase,  $R$  é a resistência em ohms,  $C$  a capacitância em farad e  $u_f$  a saída do filtro. No domínio de Laplace, a função de transferência do filtro acima fica:

$$\frac{U_f(s)}{U_d(s)} = \frac{1/RC}{s + 1/RC}. \quad (2-15)$$

Generalizando a forma do filtro, atribuindo-lhe pólos e zeros:

$$\frac{U_f(s)}{U_d(s)} = K_f \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \dots + a_1s + a_0}{s^n + b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_1s + b_0}, \quad (2-16)$$

sendo  $K_f$  é o ganho do filtro.

Como mencionado anteriormente, o detector “charge-pump” está normalmente associado a um circuito integrador, cujo comportamento é dado pelas equações a seguir:

$$u_{d \text{ int}}(t) = K_d \int_0^t \theta_e(t) dt + U_{d \text{ int inicial}},$$

ou, no domínio de Laplace:

$$\frac{U_{d \text{ int}}(s)}{\Theta_e(s)} = \frac{K_d}{s}.$$

No caso aqui estudado, propõe-se substituir o integrador por um circuito que permite o sincronismo com erro de fase não nulo – como nos detectores OU-Exclusivo e JK – porém mantendo a amplitude da faixa de operação. Tomando como base o circuito da Figura 2-8, sugere-se a modificação mostrada na Figura 2-9.

Os resistores  $R'$  foram incluídos para fazer com que o terceiro estado do detector – de alta impedância – faça com que o capacitor do filtro não mantenha a sua carga anterior. Assumindo que a frequência de entrada seja igual à frequência natural do DPLL, a saída do detector estará em seu terceiro estado, fazendo com que a saída do filtro seja  $V_{cc}/2$ , que é exatamente a tensão que provoca no oscilador sua frequência natural. O resistência  $R'$  deve ser muito menor que  $R$ , para que influa minimamente na constante de tempo do filtro, mantendo aproximadamente a função de transferência dada na equação (2-15).

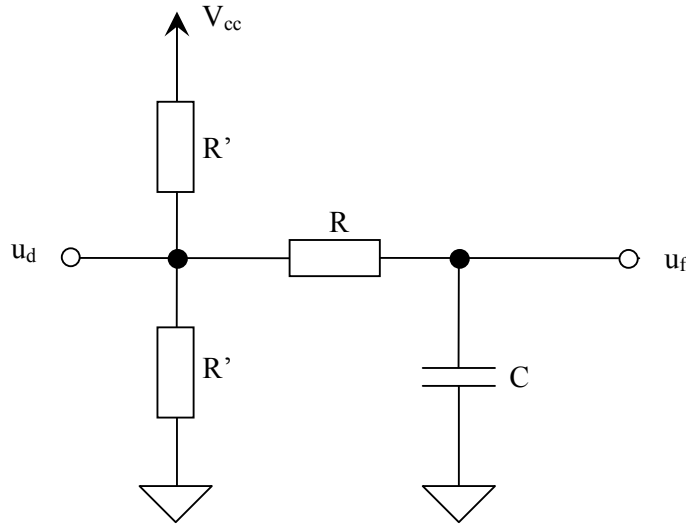


Figura 2-9: Filtro de primeira ordem modificado para adequação ao detector “charge-pump”.

Com esse novo filtro, o detector “charge-pump” adquire um comportamento semelhante ao detector JK, porém com o dobro do tamanho da faixa de operação. Além disso, pode-se considerar para o filtro a função de transferência generalizada dada pela equação (2-16).

## 2.3 O Oscilador Controlado por Tensão

O oscilador controlado por tensão – ou VCO – é um circuito que gera uma forma de onda quadrada cuja frequência varia de acordo com o sinal de entrada. Sua forma de onda é definida como:

$$v_{out}(t) = A \cdot \text{sgn}[\cos[2\pi f_o t + \varphi_{out}(t)]], \quad (2-17)$$

em que

$$\text{sgn}(\sigma) = \begin{cases} 1 & \text{se } \sigma \geq 0 \\ -1 & \text{se } \sigma < 0 \end{cases}. \quad (2-18)$$

A fase do sinal  $\varphi_{out}(t)$  é dada por:

$$\varphi_{out}(t) = 2\pi K_o \int_0^t u_f(\tau) d\tau + \varphi_{out_{inicial}}, \quad (2-19)$$

sendo  $K_o$  o ganho do VCO e  $u_f$  a saída do filtro. A resposta completa do oscilador – dada por  $\theta_{out}(t)$  – pode ser escrita como:



$$\theta_{out}(t) = 2\pi f_o t + 2\pi \int_0^t K_o u_f(\tau) d\tau + \varphi_{out\,inicial}; \quad (2-20a)$$

$$\dot{\theta}_{out}(t) = 2\pi f_o \Gamma(t) + 2\pi K_o u_f(t), \quad (2-20b)$$

sendo  $\Gamma(t)$  o degrau unitário, definido da seguinte forma:

$$\Gamma(t) = \begin{cases} 0 & \text{para } t < 0 \\ 1 & \text{para } t \geq 0 \end{cases}. \quad (2-21)$$

No domínio de Laplace, pode-se escrever a expressão da fase do  $\theta_{out}$  como:

$$\Theta_{out}(s) = \frac{2\pi K_o}{s} U_f(s) + \frac{2\pi f_o}{s}. \quad (2-22)$$

Note que a frequência natural  $f_o$  do VCO foi representada na equação (2-22) como uma entrada tipo degrau. Essa representação será adotada no modelamento da rede no espaço de estados e será independente para cada nó, o que possibilitará aplicar diferentes valores de frequência aos PLLs.

Na prática, a excursão de frequência possui um limite máximo e um mínimo, imposto pelos níveis que o detector de fase e, conseqüentemente, o filtro podem alcançar, além, é claro, do limite físico do próprio dispositivo oscilador. O conjunto de frequências localizadas dentro desses limites é conhecido como *faixa de captura*. A frequência central do oscilador está localizada no ponto médio dessa faixa.

## 2.4 O Modelo Completo do DPLL

Nos itens anteriores foram vistos, separadamente, o modelo de cada um dos blocos do DPLL, o que permite redesenhar a Figura 2-1 como segue.

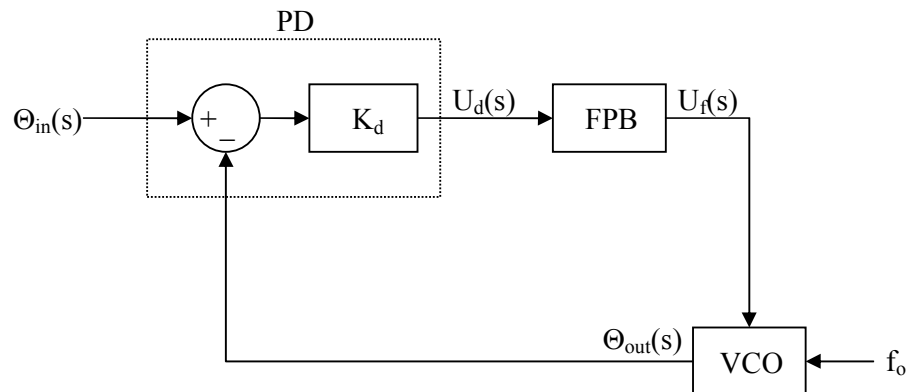


Figura 2-10: Modelo linear em blocos do DPLL.

Da equação (2-16):

$$\frac{U_f(s)}{U_d(s)} = K_f \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \dots + a_1s + a_0}{s^n + b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_1s + b_0}. \quad (2-23)$$

Com (2-7), reescreve-se a equação anterior como:

$$U_f(s) = K_f K_d \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \dots + a_1s + a_0}{s^n + b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_1s + b_0} \Theta_e(s). \quad (2-24)$$

O sinal de referência  $f_o$  da figura anterior tem por finalidade definir a frequência central do dispositivo. A referência, juntamente com o sinal do filtro, definem a frequência de oscilação do VCO. Portanto, baseado em (2-22), tem-se:

$$\Theta_{out}(s) = \frac{2\pi K_o K_f K_d}{s} \frac{a_{n-1}s^{n-1} + a_{n-2}s^{n-2} + \dots + a_1s + a_0}{s^n + b_{n-1}s^{n-1} + b_{n-2}s^{n-2} + \dots + b_1s + b_0} \Theta_e(s) + \frac{2\pi f_o}{s} \quad (2-25)$$

Sabendo-se que valem as igualdades:

$$\begin{aligned} \theta_e(t) &= \theta_{in}(t) - \theta_{out}(t); \\ \Theta_e(s) &= \Theta_{in}(s) - \Theta_{out}(s), \end{aligned} \quad (2-26)$$

a equação (2-25) reescrita no domínio do tempo fica:

$$\begin{aligned} \theta_e^{(n+1)} + b_{n-1}\theta_e^{(n)} + \dots + b_0\theta_e^{(1)} + 2\pi K[a_{n-1}\theta_e^{(n-1)} + a_{n-2}\theta_e^{(n-2)} + \dots + a_0\theta_e] &= \\ = \theta_{in}^{(n+1)} + b_{n-1}\theta_{in}^{(n)} + \dots + b_0\theta_{in}^{(1)} - 2\pi[f_o^{(n)} + b_{n-1}f_o^{(n-1)} + \dots + b_0f_o]; \end{aligned} \quad (2-27)$$

$$\begin{aligned} \theta_{out}^{(n+1)} + b_{n-1}\theta_{out}^{(n)} + \dots + b_0\theta_{out}^{(1)} + 2\pi K[a_{n-1}\theta_{out}^{(n-1)} + a_{n-2}\theta_{out}^{(n-2)} + \dots + a_0\theta_{out}] &= \\ = 2\pi K[a_{n-1}\theta_{in}^{(n-1)} + a_{n-2}\theta_{in}^{(n-2)} + \dots + a_0\theta_{in}] + 2\pi[f_o^{(n)} + b_{n-1}f_o^{(n-1)} + \dots + b_0f_o], \end{aligned} \quad (2-28)$$

com  $(\cdot)^{(i)} = \frac{d^i}{dt^i}(\cdot)$ ,  $K=K_oK_fK_d$  e  $\theta^{(i)} = \theta^{(i)}(t)$ ;  $i = 0, 1, 2, 3, \dots$  e  $t \in \mathfrak{R}$ , e dado que  $f_o$  é constante:

$$\begin{aligned} \theta_e^{(n+1)} + b_{n-1}\theta_e^{(n)} + \dots + b_0\theta_e^{(1)} + 2\pi K[a_{n-1}\theta_e^{(n-1)} + a_{n-2}\theta_e^{(n-2)} + \dots + a_0\theta_e] &= \\ = \theta_{in}^{(n+1)} + b_{n-1}\theta_{in}^{(n)} + \dots + b_0\theta_{in}^{(1)} - 2\pi b_0 f_o; \end{aligned} \quad (2-29)$$

$$\begin{aligned} \theta_{out}^{(n+1)} + b_{n-1}\theta_{out}^{(n)} + \dots + b_0\theta_{out}^{(1)} + 2\pi K[a_{n-1}\theta_{out}^{(n-1)} + a_{n-2}\theta_{out}^{(n-2)} + \dots + a_0\theta_{out}] &= \\ = 2\pi K[a_{n-1}\theta_{in}^{(n-1)} + a_{n-2}\theta_{in}^{(n-2)} + \dots + a_0\theta_{in}] + 2\pi b_0 f_o. \end{aligned} \quad (2-30)$$

Assim, as equações (2-29) e (2-30) mostram a relação entre a fase de entrada  $\theta_{in}$ , a fase de saída  $\theta_{out}$  e o erro de fases  $\theta_e$ .

## 2.4.1 Sistema Reduzido

Neste trabalho, utiliza-se para o filtro apenas os índices  $a_0$ ,  $a_1$ ,  $b_0$ ,  $b_1$ ,  $b_2$  e  $K_f$ . Os demais serão considerados nulos. Abaixo mostramos os modelos utilizando filtros de segunda ordem, com  $b_2 = 1$ . Portanto, a equação (2-29) reduz-se a:

$$\begin{aligned} \theta_e^{(3)} + b_1\theta_e^{(2)} + [b_0 + 2\pi K a_1]\theta_e^{(1)} + 2\pi K a_0\theta_e &= \\ = \theta_{in}^{(3)} + b_1\theta_{in}^{(2)} + b_0\theta_{in}^{(1)} - 2\pi b_0 f_o, \end{aligned} \quad (2-31)$$

e a equação (2-30) fica:

$$\begin{aligned} \theta_{out}^{(3)} + b_1\theta_{out}^{(2)} + [b_0 + 2\pi K a_1]\theta_{out}^{(1)} + 2\pi K a_0\theta_{out} &= \\ = 2\pi K [a_1\theta_{in}^{(1)} + a_0\theta_{in}] + 2\pi b_0 f_o. \end{aligned} \quad (2-32)$$

No domínio de Laplace, a relação entre as fases de saída e entrada e a frequência central fica:

$$\begin{aligned} \Theta_{out}(s) = \frac{a_1 s + a_0}{s^3 + b_1 s^2 + [b_0 + 2\pi K a_1]s + 2\pi K a_0} \Theta_{in}(s) + \\ + \frac{2\pi b_0 f_o}{s^3 + b_1 s^2 + [b_0 + 2\pi K a_1]s + 2\pi K a_0}. \end{aligned} \quad (2-33)$$

Com um filtro dotado de um zero e dois pólos e o VCO agindo como integrador, o DPLL torna-se um sistema de terceira ordem. No caso em que os filtros são de primeira ordem, tem-se nós de segunda ordem e adota-se  $b_2 = 0$ ,  $b_1 = 1$  e  $a_1 = 0$ .

Tendo obtido o modelo completo do DPLL, pode-se iniciar a construção dos modelos de rede no espaço de estados, para posterior discretização.

## Capítulo 3

### As Redes Mutuamente Conectadas<sup>6</sup>

Nas redes mutuamente conectadas, cada nó recebe o sinal dos demais nós, de modo a sincronizar seu oscilador. A Figura 3-1 mostra um exemplo desse tipo de topologia, com uma rede de três nós.

Como na rede mutuamente conectada cada nó necessita que mais de um dispositivo seja ligado à entrada, adota-se a *ponderação de fase*. Tomando como exemplo o nó 1, ele recebe os sinais dos nós 2 e 3 e efetua uma ponderação de fase, gerando um valor que irá controlar a fase e a frequência do oscilador 1.

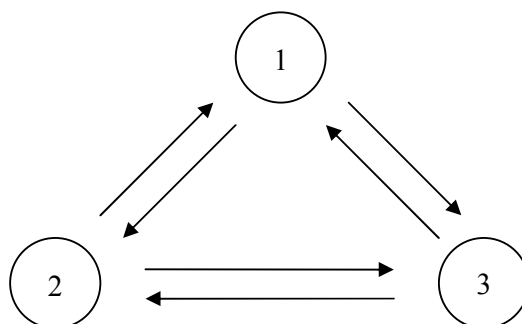


Figura 3-1: Topologia de uma rede com três nós mutuamente sincronizados e totalmente conectados.

---

<sup>6</sup> ([1] a [5])

### 3.1 Ponderação de Fase

Antes de explicar o que é a ponderação de fase, é preciso antes considerar um DPLL com mais de uma entrada. Na Figura 3-1, cada um dos DPLLs recebe o sinal de outros dois dispositivos. Portanto, nesse caso, cada DPLL deve ter duas entradas. A ponderação de fase mencionada no início do capítulo é simplesmente a média ponderada dos valores de fase na entrada do DPLL. Em termos práticos, isto pode ser feito utilizando-se dois ou mais detectores, cujas saídas estão ligadas a um somador. As entradas desse somador podem ter ganhos de sinal distintos ou idênticos, de modo que a soma de todos os ganhos seja igual a 1. A Figura 3-2 mostra um exemplo de DPLL com duas entradas ponderadas.

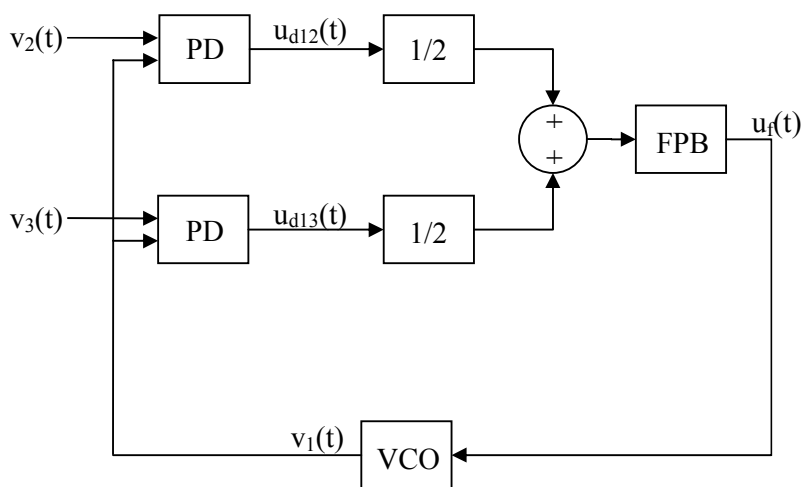


Figura 3-2: DPLL com duas entradas ponderadas.

A construção de um somador com ganhos é factível na prática e, portanto, pode-se generalizar o exemplo anterior para vários nós, com cada nó contendo diversas entradas. Essa generalização é ilustrada pela Figura 3-3, com o DPLL  $j$  dotado de  $(N-1)$  entradas ponderadas, sendo  $N$  o número de nós da rede. Os ganhos de cada detector estão multiplicados por  $1/(N-1)$ , padrão que será adotado daqui em diante. Ainda na Figura 3-3, o valor de  $u_j(t)$  fica sendo, portanto, a saída ponderada do filtro. As saídas ponderadas dos nós será utilizada para expressar os resultados das simulações mais adiante.

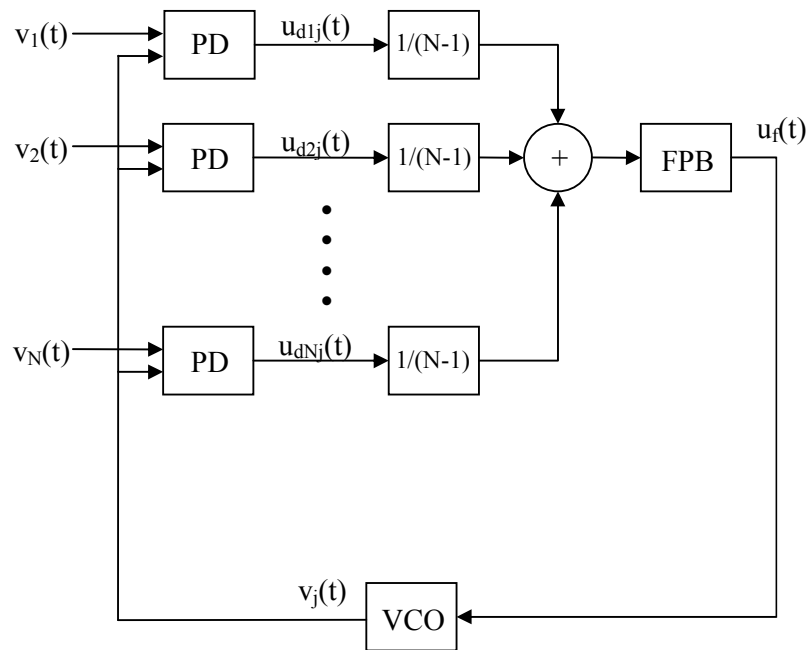


Figura 3-3: DPLL com  $N-1$  entradas ponderadas.

## 3.2 Redes com Três Nós

Determinam-se, aqui, as equações para uma rede com três nós separados fisicamente. Como daqui em diante as conexões entre os nós serão tratadas, os atrasos de percurso dos sinais farão parte do equacionamento das redes, resultado da separação entre um nó e outro. Além disso, deve-se considerar nas equações que cada DPLL é dotado de duas entradas ponderadas, como visto no item anterior.

O equacionamento da rede com três nós é importante pois, a partir dele, equacionam-se as redes com maior número de nós.

### 3.2.1 Modelo no Domínio do Tempo

As equações obtidas a seguir serão úteis para determinar o estado síncrono da rede, servindo para a definição de frequência de sincronismo.

Baseado na topologia mostrada na Figura 3-1, atribui-se um número a cada DPLL para separar as equações. Partindo das equações determinadas no capítulo anterior – item 2.4.1 – e da Figura 3-3, tem-se o seguinte modelo para o DPLL 1:

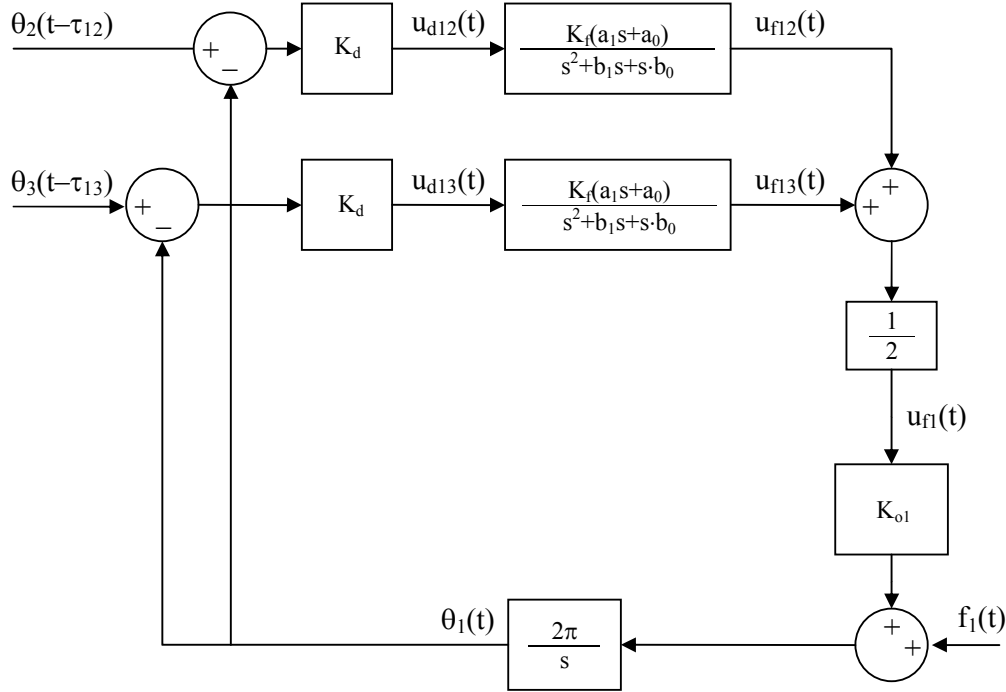


Figura 3-4: Modelo do DPLL com duas entradas ponderadas.

O DPLL 1 recebe os sinais dos dispositivos 2 e 3 atrasados no tempo. Tais atrasos são representados pelas constantes  $\tau_{12}$  e  $\tau_{13}$ . De forma similar, os outros DPLL's recebem sinais atrasados dos demais dispositivos. De acordo com a Figura 3-4:

$$\begin{aligned}
 \theta_1^{(3)}(t) = & -b_1\theta_1^{(2)}(t) - [b_0 + 2\pi K_1 a_1] \theta_1^{(1)}(t) - 2\pi K_1 a_0 \theta_1(t) + \\
 & + \frac{2\pi K_1}{2} [a_1 \theta_2^{(1)}(t - \tau_{12}) + a_0 \theta_2(t - \tau_{12})] + \\
 & + \frac{2\pi K_1}{2} [a_1 \theta_3^{(1)}(t - \tau_{13}) + a_0 \theta_3(t - \tau_{13})] + \\
 & + 2\pi b_0 f_1,
 \end{aligned} \tag{3-1}$$

sendo  $K_l = K_d K_f K_{o1}$  e  $f_l$  a frequência central do DPLL 1. Vale notar que, nesse caso,  $u_{fl}(t)$  representa a saída ponderada dos filtros do nó 1, equivalente a  $u_f(t)$  na Figura 3-3, sem perda de generalidade.

São mantidas idênticas todas as constantes em todos os DPLLs, exceto  $K_o$ , que no caso do DPLL 1 é  $K_{o1}$ . De modo similar, os modelos dos demais dispositivos são dados por:

$$\begin{aligned}
\theta_2^{(3)}(t) = & -b_1\theta_2^{(2)}(t) - [b_0 + 2\pi K_2 a_1]\theta_2^{(1)}(t) - 2\pi K_2 a_0\theta_2(t) + \\
& + \frac{2\pi K_2}{2}[a_1\theta_1^{(1)}(t - \tau_{21}) + a_0\theta_1(t - \tau_{21})] + \\
& + \frac{2\pi K_2}{2}[a_1\theta_3^{(1)}(t - \tau_{23}) + a_0\theta_3(t - \tau_{23})] + \\
& + 2\pi b_0 f_2;
\end{aligned} \tag{3-2}$$

$$\begin{aligned}
\theta_3^{(3)}(t) = & -b_1\theta_3^{(2)}(t) - [b_0 + 2\pi K_3 a_1]\theta_3^{(1)}(t) - 2\pi K_3 a_0\theta_3(t) + \\
& + \frac{2\pi K_3}{2}[a_1\theta_1^{(1)}(t - \tau_{31}) + a_0\theta_1(t - \tau_{31})] + \\
& + \frac{2\pi K_3}{2}[a_1\theta_2^{(1)}(t - \tau_{32}) + a_0\theta_2(t - \tau_{32})] + \\
& + 2\pi b_0 f_3;
\end{aligned} \tag{3-3}$$

sendo, analogamente ao primeiro PLL,  $K_2 = K_d K_f K_{o2}$  e  $K_3 = K_d K_f K_{o3}$ .  $f_2$  e  $f_3$  são as frequências naturais dos dispositivos 2 e 3, respectivamente.

Antes de prosseguir, é necessário esclarecer que os atrasos entre dois nós é o mesmo tanto na ida, quanto na volta. Isto é, considerando os PLLs 1 e 2,  $\tau_{12} = \tau_{21}$ .

Vale ressaltar ainda que a saída do filtro foi normalizada, ou seja, com o PLL em “lock” na frequência natural e erro de fase nulo, o valor na saída do filtro é igual a zero.

### 3.3 Redes com N nós

A partir daqui, os resultados obtidos nas equações (3-1), (3-2) e (3-3) serão generalizados para uma rede com  $N$  nós. Assim, para um nó  $j$  qualquer:

$$\begin{aligned}
\theta_j^{(3)}(t) = & -b_1\theta_j^{(2)}(t) - [b_0 + 2\pi K_j a_1]\theta_j^{(1)}(t) - 2\pi K_j a_0\theta_j(t) + \\
& + \frac{2\pi K_j}{N-1} \sum_{m \neq j}^N [a_1\theta_m^{(1)}(t - \tau_{jm}) + a_0\theta_m(t - \tau_{jm})] \\
& + 2\pi b_0 f_j,
\end{aligned} \tag{3-4}$$

com  $K_j = K_d K_f K_{oj}$  e  $f_j$  a frequência central do DPLL  $j$ . Partindo de (3-4), obtém-se uma solução para o estado síncrono da rede. A estabilidade desse estado de equilíbrio será analisada após a definição do modelo contínuo da rede e sua discretização.

#### 3.3.1 Estado Síncrono

Para encontrar o estado síncrono – representado por um ponto de equilíbrio – utiliza-se a equação generalizada de (3-4), fazendo algumas considerações.

Os sinais de referência determinam a frequência central ou natural do PLL. Esta é a frequência para uma saída nula do filtro. Pode-se pensar nas referências como sinais tipo



degrau cujos níveis vão determinar as frequências centrais em cada PLL. Em regime,  $f_j$  é constante e  $f_j^{(1)}$  e  $f_j^{(2)}$  são nulos. Portanto,  $\omega_j = 2\pi f_j$  é a frequência natural do nó  $j$ , dada em radianos por segundo.

Em torno do estado síncrono, no que se refere às fases dos sinais,  $\theta^{(1)}$  é igual a  $\omega_s$ , que é a frequência de sincronismo da rede.  $\theta^{(2)}$  e  $\theta^{(3)}$  são nulos, pois não há variação em  $\omega_s$ . Além disso, vale a seguinte aproximação:

$$\begin{aligned}\theta(t) &\cong \omega_s t + \varphi; \\ \theta(t - \tau) &\cong \theta(t) - \omega_s \tau.\end{aligned}\quad (3-5)$$

Do que foi exposto e de (3-5), para o DPLL  $j$ :

$$\begin{aligned}0 &= -[b_0 + 2\pi K_j a_1] \omega_s - 2\pi K_j a_0 \theta_j(t) + \frac{2\pi K_j}{N-1} \sum_{\substack{m \\ m \neq j}}^N [a_1 \omega_s + a_0 \theta_m(t) - a_0 \omega_s \tau_{jm}] \\ &\quad + 2\pi b_0 r_j; \\ 0 &= -b_0 \omega_s - 2\pi K_j a_0 \theta_j(t) + \frac{2\pi K_j}{N-1} \sum_{\substack{m \\ m \neq j}}^N [a_0 \theta_m(t) - a_0 \omega_s \tau_{jm}] + b_0 \omega_j.\end{aligned}\quad (3-6)$$

Dividindo a equação (3-6) por  $K_j$  e  $b_0$  e somando-a uma a uma para cada nó  $j$  da rede, os termos  $\theta_m$  e  $\theta_j$  cancelam-se, resultando:

$$0 = -\omega_s \sum_{j=1}^N \left[ \frac{1}{K_j} \right] - \frac{4\pi a_0 \omega_s}{b_0 (N-1)} \sum_{i=1}^M [\tau_i] + \sum_{j=1}^N \left[ \frac{\omega_j}{K_j} \right]. \quad (3-7)$$

Na equação (3-7), a notação  $\tau_i$ , com  $i=1,2,\dots,M$ , foi adotada pois, tomando-se um par de nós  $i$  qualquer, assume-se que os atrasos de ida e volta entre os nós desse par são idênticos.  $M$  é o número total de pares de nós da rede. Para uma rede mutuamente conectada, o valor de  $M$  pode ser determinado pela combinação de  $N$  nós, dois a dois:

$$M = \frac{N!}{2 \cdot (N-2)!}.$$

Numa rede de quatro nós, por exemplo, existe um total de seis atrasos –  $M = 6$  – e seus valores são  $\tau_1, \dots, \tau_6$ . Esses atrasos, como será visto adiante, estão associados aos ganhos do modelo no espaço de estados de tempo contínuo  $A_{c1}$  a  $A_{c6}$ .

Tomando agora (3-7) e isolando  $\omega_s$ , obtém-se a frequência de sincronismo para a rede com atrasos, como:

$$\omega_s = \frac{\sum_{j=1}^N \left[ \frac{\omega_j}{K_j} \right]}{\sum_{j=1}^N \left[ \frac{1}{K_j} \right] + \frac{4\pi a_0}{b_0 (N-1)} \sum_{i=1}^M [\tau_i]}.\quad (3-8)$$

De posse do estado síncrono da rede, deseja-se verificar sua estabilidade para uma rede qualquer de  $N$  nós. Um detalhe importante a ser notado é a não distinção de que tipo de detector de fase utilizado. Todos os comparadores citados apresentam um comportamento próximo do linear sob certas condições, diferindo apenas no ganho, embutido na constante  $K_j$ . Dessa forma, a equação (3-8) torna-se válida, independentemente do tipo de comparador.

Outro detalhe importante é que os únicos parâmetros do filtro que exercem influência na igualdade acima são  $a_0$  e  $b_0$ , mesmo que haja um maior número de pólos e zeros no filtro. De fato, são estes os termos que determinam o estado do regime estacionário do filtro.

A equação do sincronismo calculada aqui é semelhante à obtida em (1) para o PLL analógico. A diferença está no fato de que (3-8) expõe a distinção entre as constantes que servem como parâmetros para os dispositivos.

### 3.4 Representação no Espaço de Estados

O modelo no espaço de estados será importante daqui para frente pois, fundamentado nele, será obtido o modelo discreto da rede de sincronismo. Existem várias formas de se construir modelos no espaço de estados – as chamadas *realizações*. Aqui, os estados são compostos pelos valores das fases e dos estados dos filtros. Para tanto, o equacionamento no domínio do tempo que faremos difere do visto no item 3.3.1.

Por hora, considera-se apenas o modelo do DPLL  $j$ . Primeiramente, por se tratar de um sistema linear, representa-se o filtro deste PLL no espaço de estados por:

$$\begin{aligned} \dot{x}_{fj} &= A_{fj}x_{fj} + B_{fj}u_{dj}; \\ u_{fj} &= G_{fj}x_{fj}, \end{aligned} \quad (3-9)$$

com  $u_{dj}$  sendo a entrada do sistema.

O filtro é alimentado pelos valores das diferenças de fase ponderadas fornecidos pelos detectores de fase. Portanto:

$$u_{dj} = K_d \sum_{\substack{m \\ m \neq j}}^N \frac{\theta_m(t - \tau_{jm})}{N-1} - K_d \theta_j(t). \quad (3-10)$$

Recordando a equação (2-20b) do VCO, fazendo as devidas adaptações:

$$\dot{\theta}_j(t) = 2\pi f_j \Gamma(t) + 2\pi K_{oj} u_{fj}(t), \quad (3-11)$$

implicando a construção da primeira representação de estados do DPLL  $j$ :

$$\begin{bmatrix} \dot{\theta}_j \\ \dot{x}_{fj} \end{bmatrix} = \begin{bmatrix} 0 & 2\pi K_{oj} G_{fj} \\ -B_{fj} K_d & A_{fj} \end{bmatrix} \begin{bmatrix} \theta_j \\ x_{fj} \end{bmatrix} + \begin{bmatrix} 2\pi f_j \\ 0 \end{bmatrix} \Gamma(t) + \begin{bmatrix} 0 \\ B_{fj} \end{bmatrix} K_d \sum_{\substack{m \\ m \neq j}}^N \frac{\theta_m(t - \tau_{jm})}{N-1}. \quad (3-12)$$

Observando a equação (3-12), nota-se que as fases dos demais nós da rede fazem parte da representação no espaço de estados, o que era esperado, já que cada dispositivo está interligado com os demais. Devido ao atraso entre os nós, essas fases atrasadas excitam o sistema como novas entradas. Elas são representadas como estados do sistema e, conseqüentemente, a dimensão do espaço de estados fica maior. Define-se, então, o seguinte sistema contínuo:

$$\begin{aligned}\dot{x}(t) &= A_c x(t) + B_c u(t) + \sum_{i=1}^M A_{ci} x(t - \tau_i); \\ y(t) &= G_c x(t).\end{aligned}\quad (3-13)$$

A exemplo do que foi feito na equação (3-7), os atrasos entre dois nós consecutivos estão representados por um mesmo símbolo. O sistema (3-13) fornece um modo de representar os estados com e sem atrasos, por meio de  $x(t)$  e  $x(t - \tau_i)$ .  $A_c$  e  $A_{ci}$  são as matrizes de ganho dos estados;  $B_c$  é a matriz de ganho da entrada;  $G_c$  é a matriz de ganhos da saída.

Esse sistema generalizado descreve o comportamento global da rede, agregando o comportamento dinâmico de todos os DPLLs – representados individualmente pela equação (3-12) – e suas conexões. No próximo capítulo, um método para discretização da equação (3-13) será desenvolvido, para estudar a estabilidade da malha de sincronismo.

A seguir, um exemplo para uma rede de três nós é mostrado, a fim de exemplificar o método de construção de uma rede no espaço de estados. A forma apresentada pode ser utilizada para sistemas de maior dimensão.

### 3.4.1 Representação de uma Rede de Três Nós

Adota-se, aqui, um filtro de segunda ordem para todos os PLLs, cuja função de transferência é:

$$\frac{U_f(s)}{U_d(s)} = \frac{a_1 s + a_0}{s^2 + b_1 s + b_0}. \quad (3-14)$$

Modificando a interpretação do filtro para o espaço de estados, tem-se:

$$\begin{aligned}\begin{bmatrix} \dot{x}_f \\ \ddot{x}_f \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -b_0 & -b_1 \end{bmatrix} \begin{bmatrix} x_f \\ \dot{x}_f \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_d; \\ u_f &= \begin{bmatrix} a_0 & a_1 \end{bmatrix} \begin{bmatrix} x_f \\ \dot{x}_f \end{bmatrix}.\end{aligned}\quad (3-15)$$

Esse filtro será utilizado nos três nós com os mesmos parâmetros. Os atrasos entre dispositivos, como exposto anteriormente, é o mesmo em ambos os sentidos, seguindo a notação:

$$\begin{aligned}
\tau_{12} &= \tau_{21} = \tau_1; \\
\tau_{13} &= \tau_{31} = \tau_2; \\
\tau_{23} &= \tau_{32} = \tau_3.
\end{aligned} \tag{3-16}$$

Portanto, tomando o nó 1 e utilizando a equação (3-12):

$$\begin{aligned}
\begin{bmatrix} \dot{\theta}_1 \\ \dot{x}_{f1} \\ \ddot{x}_{f1} \end{bmatrix} &= \begin{bmatrix} 0 & 2\pi K_{o1} a_0 & 2\pi K_{o1} a_1 \\ 0 & 0 & 1 \\ -K_d & -b_0 & -b_1 \end{bmatrix} \begin{bmatrix} \theta_1 \\ x_{f1} \\ \dot{x}_{f1} \end{bmatrix} + \begin{bmatrix} 2\pi f_1 \\ 0 \\ 0 \end{bmatrix} \Gamma(t) \\
&+ \begin{bmatrix} 0 \\ 0 \\ K_d/2 \end{bmatrix} (\theta_2(t - \tau_1) + \theta_3(t - \tau_2)).
\end{aligned} \tag{3-17}$$

Assume-se que o ganho do detector de fase também é o mesmo para todos os DPLLs e passando para o formato definido pela equação (3-13), especifica-se o estado completo da rede como:

$$x(t) = [\theta_1(t) \quad x_{f1}(t) \quad \dot{x}_{f1}(t) \quad \theta_2(t) \quad x_{f2}(t) \quad \dot{x}_{f2}(t) \quad \theta_3(t) \quad x_{f3}(t) \quad \dot{x}_{f3}(t)]^T. \tag{3-18}$$

Finalmente, determinam-se as matrizes de ganho do sistema, começando por  $A_c$ :

$$A_c = \begin{bmatrix} 0 & 2\pi K_{o1} a_0 & 2\pi K_{o1} a_1 & & & & & & \\ 0 & 0 & 1 & & 0_{3 \times 3} & & & & 0_{3 \times 3} \\ -K_d & -b_0 & -b_1 & & & & & & \\ & & & 0 & 2\pi K_{o2} a_0 & 2\pi K_{o2} a_1 & & & \\ & 0_{3 \times 3} & & 0 & 0 & 1 & & & 0_{3 \times 3} \\ & & & -K_d & -b_0 & -b_1 & & & \\ & & & & & & 0 & 2\pi K_{o3} a_0 & 2\pi K_{o3} a_1 \\ & 0_{3 \times 3} & & & 0_{3 \times 3} & & 0 & 0 & 1 \\ & & & & & & -K_d & -b_0 & -b_1 \end{bmatrix}, \tag{3-19}$$

sendo  $0_{3 \times 3}$  a matriz nula  $3 \times 3$ . O ganho de entrada  $B_c$  fica:

$$B_c = \begin{bmatrix} 2\pi & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\pi & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2\pi & 0 & 0 \end{bmatrix}^T, \tag{3-20}$$

lembrando que as entradas do sistema são do tipo degrau:

$$u(t) = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \end{bmatrix} \Gamma(t). \tag{3-21}$$

Segundo (3-16), tem-se três valores de atrasos possíveis e, desse modo, três ganhos associados às distâncias a serem vencidas,  $A_{c1}$ ,  $A_{c2}$  e  $A_{c3}$ :

$$A_{c1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0_{3 \times 3} & 0 & 0 & 0 & 0_{3 \times 3} \\ & & K_d/2 & 0 & 0 \\ & & & & \\ 0 & 0 & 0 & 0_{3 \times 3} & 0_{3 \times 3} \\ K_d/2 & 0 & 0 & & \\ 6,1 & & & & \\ & 0_{3 \times 3} & & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad (3-22)$$

$$A_{c2} = \begin{bmatrix} 0 & 0 & 0 \\ 0_{3 \times 3} & 0_{3 \times 3} & 0 & 0 & 0 \\ & & K_d/2 & 0 & 0 \\ & & & & \\ 0_{3 \times 3} & 0_{3 \times 3} & & 0_{3 \times 3} & \\ 0 & 0 & 0 & 0_{3 \times 3} & 0_{3 \times 3} \\ K_d/2 & 0 & 0 & & \\ 9,1 & & & & \end{bmatrix}; \quad (3-23)$$

$$A_{c3} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0 & 0 & 0 \\ & & K_d/2 & 0 & 0 \\ & & & & \\ 0_{3 \times 3} & 0 & 0 & 0 & 0_{3 \times 3} \\ & K_d/2 & 0 & 0 & \\ & 9,4 & & & \end{bmatrix}. \quad (3-24)$$

Na construção de  $A_{c1}$ , o ganho (3-22) refere-se ao estado atrasado  $x(t-\tau_1)$  e, dessa maneira, mostra as conexões entre os nós 1 e 2.  $A_{c1}$  foi dividida em blocos  $3 \times 3$  para facilitar esta interpretação. O primeiro bloco, de cima para baixo, em que aparece a contante  $K_d/2$  representa a conexão que vai de 2 a 1. A posição dessa constante dentro da matriz é (3;4) e leva o valor da fase  $\theta_2(t-\tau_1)$  até a entrada do filtro do nó 1. Do mesmo modo, a outra constante está localizada em (6;1) e pertence ao bloco que conecta 1 a 2. Nessa posição,  $K_d/2$  aplica  $\theta_1(t-\tau_1)$  à entrada do nó 2. Esta construção é válida para as matrizes de atraso  $A_{c2}$  e  $A_{c3}$ , ressaltando que  $A_{c2}$  reflete as conexões entre os nós 1 e 3 e a matriz  $A_{c3}$  entre os nós 2 e 3.

Para a saída, pode-se escolher quaisquer elementos do estado em quaisquer configurações, já que a saída funcionará apenas como elemento de observação, e não como parte de qualquer

malha de controle ou estabilização por realimentação. No capítulo que contém os resultados das simulações, por exemplo, será dada ênfase à resposta dos filtros, sendo conveniente apresentá-la na saída de nosso sistema. Assim, podemos escrever  $G_c$  como:

$$G_c = \begin{bmatrix} 0 & a_0 & a_1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a_0 & a_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_0 & a_1 \end{bmatrix}. \quad (3-25)$$

Por fim, baseado em (3-13), a representação completa da malha de sincronismo de três nós fica da seguinte forma:

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t) + A_{c1} x(t - \tau_1) + A_{c2} x(t - \tau_2) + A_{c3} x(t - \tau_3); \\ y(t) &= G_c x(t). \end{aligned} \quad (3-26)$$

No próximo capítulo será apresentado um método para discretização de sistemas como (3-26). O apêndice B fornece a listagem do programa em MATLAB chamado CRIAREDE para gerar o modelo contínuo, de acordo com a realização mostrada aqui.

## Capítulo 4

### Discretização<sup>7</sup>

Ao longo deste capítulo, um método para discretização de sistemas contínuos no tempo e dotados de atrasos é apresentado. Aqui completa-se o processo que se iniciou nos capítulos anteriores com o modelamento do DPLL, e passou pelo modelamento da rede de sincronismo no espaço de estados.

O intuito de se representar sistemas com atrasos é obter uma vantagem analítica. Com o modelo dinâmico discretizado, espera-se transformar as entradas e estados do tipo  $u(t-\tau)$  e  $x(t-\tau)$  em funções discretas, visto que no modelo contínuo o tratamento destes termos é analiticamente mais complicado.

No caso estudado, é possível avaliar a estabilidade das malhas de PLLs em torno dos pontos de equilíbrio por meio da análise dos autovalores do sistema obtido com a discretização.

Inicia-se mostrando um sistema contínuo linear e invariante no tempo – LTI – típico, representado como:

$$\begin{aligned}\dot{x} &= A_c x + B_c u; \\ y &= G_c x.\end{aligned}\tag{4-1}$$

sendo  $x$  o estado,  $u$  a entrada,  $y$  a saída do sistema e  $A_c$ ,  $B_c$ ,  $G_c$  matrizes constantes.

O sistema aqui proposto tem a seguinte forma:

$$\begin{aligned}\dot{x} &= A_c x + B_c u + \sum_{i=1}^M A_{c_i} x(t - \tau_i) + \sum_{i=1}^M B_{c_i} u(t - \tau_i); \\ y &= G_c x + \sum_{i=1}^M G_{c_i} x(t - \tau_i),\end{aligned}\tag{4-2}$$

sendo  $\tau_i$  os atrasos de transmissão.  $A_{c_i}$ ,  $B_{c_i}$ ,  $G_{c_i}$  são matrizes constantes para qualquer  $i=0\dots M$ . Foram consideradas entradas e estados atrasados, a despeito do que foi feito para o modelo

---

<sup>7</sup> ([7])

contínio do capítulo anterior, com vistas a tornar o sistema genérico e flexível. O vetor de estados  $x$  é da forma  $p \times 1$ , o de entradas  $u$  é da forma  $q \times 1$  e o de saída  $y$  é  $r \times 1$ . Quanto às matrizes,  $A_c$  e  $A_{ci}$  são da forma  $p \times p$ ;  $B_c$  e  $B_{ci}$  são  $p \times q$ ;  $G_c$  e  $G_{ci}$  são  $r \times p$ , qualquer que seja o valor de  $i$ .

Como feito em ([7]) e ([16]), discretiza-se o sistema contínuo, partindo de sua solução geral. Portanto, dado o sistema contínuo (4-2), realimentado com estados desafiados no tempo, a solução geral fica:

$$\begin{aligned}
 x(t) = & e^{A_c(t-t_0)}x(t_0) + \int_{t_0}^t e^{A_c(t-\lambda)}B_c u(\lambda)d\lambda + \\
 & + \sum_{i=1}^M \int_{t_0}^t e^{A_c(t-\lambda)}A_{ci}x(\lambda - \tau_i)d\lambda + \\
 & + \sum_{i=1}^M \int_{t_0}^t e^{A_c(t-\lambda)}B_{ci}u(\lambda - \tau_i)d\lambda.
 \end{aligned} \tag{4-3}$$

Para mostrar que (4-3) é solução do sistema contínuo, basta derivar ambos os lados da equação.

$$\begin{aligned}
 \dot{x}(t) = & A_c e^{A_c(t-t_0)}x(t_0) + A_c e^{A_c t} \int_{t_0}^t e^{-A_c \lambda} B_c u(\lambda) d\lambda + e^{A_c t} e^{-A_c t} B_c u(t) + \\
 & + \sum_{i=1}^M \left[ A_c e^{A_c t} \int_{t_0}^t e^{-A_c \lambda} A_{ci} x(\lambda - \tau_i) d\lambda + e^{A_c t} e^{-A_c t} A_{ci} x(t - \tau_i) \right] + \\
 & + \sum_{i=1}^M \left[ A_c e^{A_c t} \int_{t_0}^t e^{-A_c \lambda} B_{ci} u(\lambda - \tau_i) d\lambda + e^{A_c t} e^{-A_c t} B_{ci} u(t - \tau_i) \right] \Rightarrow \\
 \dot{x}(t) = & A_c \left\{ e^{A_c(t-t_0)}x(t_0) + \int_{t_0}^t e^{A_c(t-\lambda)}B_c u(\lambda)d\lambda + \right. \\
 & + \sum_{i=1}^M \int_{t_0}^t e^{A_c(t-\lambda)}A_{ci}x(\lambda - \tau_i)d\lambda + \\
 & \left. + \sum_{i=1}^M \int_{t_0}^t e^{A_c(t-\lambda)}B_{ci}u(\lambda - \tau_i)d\lambda \right\} + \\
 & + B_c u(t) + \sum_{i=1}^M [A_{ci}x(t - \tau_i)] + \sum_{i=1}^M [B_{ci}u(t - \tau_i)] \Rightarrow \\
 \dot{x}(t) = & A_c x(t) + B_c u(t) + \sum_{i=1}^M [A_{ci}x(t - \tau_i)] + \sum_{i=1}^M [B_{ci}u(t - \tau_i)]
 \end{aligned}$$

Portanto, mostra-se que (4-3) é solução geral de (4-2).

A partir do sistema de tempo contínuo, deve-se obter seu equivalente de tempo discreto, definido como:



$$\begin{aligned}
x_{k+1} &= A_d x_k + B_d u_k + \sum_{i=1}^M \overset{pos}{A_{d_i}} x_{k+1-n_i} + \sum_{i=1}^M \overset{act}{A_{d_i}} x_{k-n_i} + \sum_{i=1}^M \overset{pre}{A_{d_i}} x_{k-1-n_i} + \sum_{i=1}^M \overset{act}{B_{d_i}} u_{k-n_i} + \sum_{i=1}^M \overset{pre}{B_{d_i}} u_{k-1-n_i}; \\
y_k &= G_d x_k + \sum_{i=1}^M \overset{act}{G_{d_i}} x_{k-n_i} + \sum_{i=1}^M \overset{pre}{G_{d_i}} x_{k-1-n_i}.
\end{aligned} \tag{4-4}$$

sendo  $n_i$  um número inteiro e os índices superiores *pos*, *act* e *pre* definem respectivamente os ganhos dos estados posteriores, atuais e anteriores, com relação a  $n_i$ . Substituindo  $t$  por  $(k+1)T_s$  e  $t_0$  por  $kT_s$  em (4-3), com  $T_s$  representando o período de amostragem:

$$\begin{aligned}
x((k+1)T_s) &= e^{A_c T_s} x(kT_s) + \int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s-\lambda)} B_c u(\lambda) d\lambda + \\
&+ \sum_{i=1}^n \int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s-\lambda)} A_{c_i} x(\lambda - \tau_i) d\lambda + \\
&+ \sum_{i=1}^n \int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s-\lambda)} B_{c_i} u(\lambda - \tau_i) d\lambda .
\end{aligned} \tag{4-5}$$

Deduzimos de imediato a matrizes  $A_d$  e  $C_d$ , que ficam:

$$\begin{aligned}
A_d &= e^{A_c T_s}; \\
G_d &= G_c.
\end{aligned} \tag{4-6}$$

As demais matrizes serão determinadas a seguir.

## 4.1 A Entrada

### 4.1.1 Entrada sem Atrasos

A matriz  $B_d$  será determinada considerando-se a entrada  $u$  constante dentro do intervalo  $kT_s$  e  $(k+1)T_s$  e igual a  $u(kT_s)$ . Pode-se pensar o sistema contínuo excitado por  $u$  através de um segurador de ordem zero – ZOH ([16]). Assim, tem-se:

$$\begin{aligned}
&\int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s-\lambda)} B_c u(\lambda) d\lambda = \\
&= \left[ \int_0^{T_s} e^{A_c \mu} d\mu \right] B_c u(kT_s) = \left[ \int_0^{T_s} e^{A_c \mu} d\mu \right] B_c u_k,
\end{aligned} \tag{4-7}$$

sendo  $\mu = (k+1)T_s - \lambda$ . Desse modo:

$$B_d = \left[ \int_0^{T_s} e^{A_c \mu} d\mu \right] B_c. \tag{4-8}$$

É importante notar que  $A_d$  e  $B_d$  são idênticas às matrizes obtidas em ([16]), a partir de um o sistema de tempo contínuo linear típico.

#### 4.1.2 Entrada com Atrasos

As matrizes  ${}^{act}B_{di}$  e  ${}^{pre}B_{di}$  estão associadas à mesma entrada  $u(t-\tau_i)$ , que é a entrada  $u(t)$  atrasada. Essas matrizes são obtidas a partir de  $B_{ci}$ , de modo similar ao feito para  $B_d$ :

$$\begin{aligned} & \int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s-\lambda)} B_{ci} u(\lambda - \tau_i) d\lambda = \\ & = e^{A_c T_s} \int_0^{T_s} e^{-A_c \mu} B_{ci} u(\mu + kT_s - \tau_i) d\mu, \end{aligned} \quad (4-9)$$

com  $\mu = \lambda - kT_s$ . Sendo  $\text{floor}(\sigma)$  a função que retorna o maior número inteiro menor ou igual que  $\sigma$ , representa-se  $\tau_i$  por:

$$\tau_i = n_i T_s + c_i; \quad n_i = \text{floor}(\tau_i); \quad c_i \in \mathfrak{R}, \quad (4-10)$$

de modo que  $0 \leq c_i < T_s$ . Os valores  $n_i$  e  $c_i$  serão utilizados mais adiante para o cálculo dos ganhos referentes aos estados defasados.

A Figura 4-1 mostra um exemplo de uma entrada atrasada no tempo, para o caso de dimensão 1. A partir dessa figura e das igualdades em (4-9) e (4-10), observa-se que, para valores de  $\mu$  entre 0 e  $c_i$ ,  $u(\mu + kT_s - \tau_i)$  assume valores iguais  $u((k-n_i-1)T_s)$ .

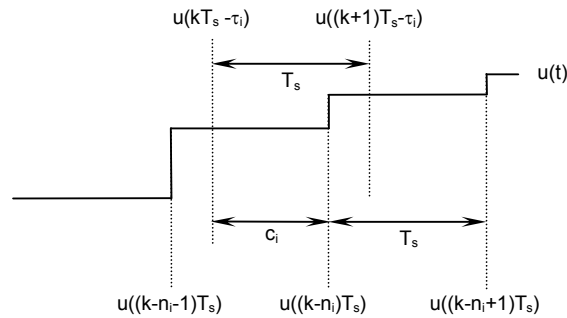


Figura 4-1: Entrada  $u(t)$  atrasada.

Quando  $\mu$  está entre  $c_i$  e  $T_s$ ,  $u(\mu + kT_s - \tau_i)$  é igual a  $u((k-n_i)T_s)$ . Portanto, a integral à direita em (4-9) fica:

$$\begin{aligned} & e^{A_c T_s} \int_0^{T_s} e^{-A_c \mu} B_{ci} u(\mu + kT_s - \tau_i) d\mu = \\ & = e^{A_c T_s} \left[ \int_0^{c_i} e^{A_c \mu} d\mu \right] B_{ci} u((k-n_i-1)T_s) + e^{A_c T_s} \left[ \int_{c_i}^{T_s} e^{A_c \mu} d\mu \right] B_{ci} u((k-n_i)T_s). \end{aligned} \quad (4-11)$$

Da equação (4-11), extrai-se as constantes de entrada:

$${}^{act}B_{di} = e^{A_c T_s} \left[ \int_{c_i}^{T_s} e^{A_c \mu} d\mu \right] B_{c_i}; \quad (4-12a)$$

$${}^{pre}B_{di} = e^{A_c T_s} \left[ \int_0^{c_i} e^{A_c \mu} d\mu \right] B_{c_i}, \quad (4-12b)$$

que são as matrizes das entradas atrasadas, enunciadas no modelo discreto proposto em (4-4). Devido à linearidade destas transformações, o resultado em (4-12) vale para qualquer valor de  $i$ .

## 4.2 A Realimentação de Estados

As matrizes de realimentação  ${}^{pos}A_{di}$ ,  ${}^{act}A_{di}$ , e  ${}^{pre}A_{di}$  serão determinadas considerando-se que os estados são funções afim. Para determinar os valores dessas integrais, a seguir será utilizada a interpolação entre estados consecutivos. Deseja-se com isso conseguir maior precisão no cálculo das integrais, considerando a realimentação de um estado com um atraso  $\tau_i$ , cujos resultados podem ser extrapolados para os demais estados atrasados, ou seja, para qualquer  $i$ .

Como feito anteriormente, substitui-se o valor  $t$  por  $(k+1)T_s$  e  $t_0$  por  $kT_s$  no termo que contém a matriz  $A_{c_i}$  em (4-3). Assim:

$$\int_{kT_s}^{(k+1)T_s} e^{A_c((k+1)T_s - \lambda)} A_{c_i} x(\lambda - \tau_i) d\lambda. \quad (4-13)$$

Com a mudança de variável  $\mu = \lambda - kT_s$ :

$$\begin{aligned} & \int_0^{T_s} e^{-A_c(\mu - T_s)} A_{c_i} x(\mu + kT_s - \tau_i) d\mu = \\ & = e^{A_c T_s} \int_0^{T_s} e^{-A_c \mu} A_{c_i} x(\mu + kT_s - \tau_i) d\mu. \end{aligned} \quad (4-14)$$

Para calcular a integral que está em (4-14), considera-se que  $x$ , no intervalo entre os tempos de amostragem  $kT_s$  e  $(k+1)T_s$ , é aproximadamente uma função afim da forma  $\alpha_i \mu + \beta_i$ , sendo  $\alpha_i$  e  $\beta_i$  vetores. Além disso, considera-se o atraso  $\tau_i$  como feito em (4-10). Em termos gráficos, tem-se algo como mostra a Figura 4-2:

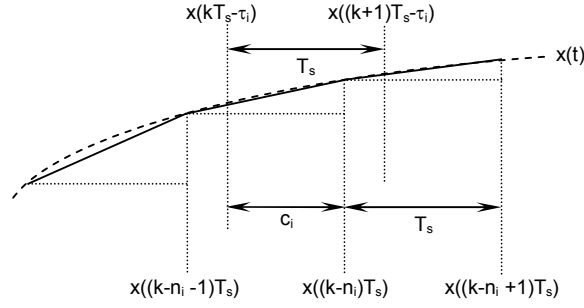


Figura 4-2: Aproximação do estado de  $x(t)$ .

De (4-10) e de (4-14), tem-se:

$$e^{A_c T_s} \int_0^{T_s} e^{-A_c \mu} A_{c_i} x(\mu + kT_s - \tau_i) d\mu = e^{A_c T_s} \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} x(\mu + (k-n_i)T_s - c_i) d\mu + \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} x(\mu + (k-n_i)T_s - c_i) d\mu \right]. \quad (4-15)$$

De acordo com a Figura 4-2 e aproximando  $x$  por uma função afim, para valores de  $\mu$  entre 0 e  $c_i$ ,  $x(\mu + kT_s - \tau_i)$  assume valores entre  $x((k-n_i-1)T_s)$  e  $x((k-n_i)T_s)$ . Já quando  $\mu$  está entre  $c_i$  e  $T_s$ ,  $x(\mu + kT_s - \tau_i)$  está entre  $x((k-n_i)T_s)$  e  $x((k-n_i+1)T_s)$ .

Visto que, para cada intervalo considerado:

$$x(t) \cong \alpha_i \mu + \beta_i, \quad (4-16)$$

as integrais à direita em (4-15) têm a forma:

$$\int_{l_{\text{inf}}}^{l_{\text{sup}}} e^{-A_c \mu} A_{c_i} (\alpha_i \mu + \beta_i) d\mu = \int_{l_{\text{inf}}}^{l_{\text{sup}}} e^{-A_c \mu} A_{c_i} \alpha_i \mu d\mu + \int_{l_{\text{inf}}}^{l_{\text{sup}}} e^{-A_c \mu} A_{c_i} \beta_i d\mu. \quad (4-17)$$

#### 4.2.1 De 0 a $c_i$

De acordo com a Figura 4-2, no intervalo de 0 a  $c_i$ , os valores de  $\alpha$  e  $\beta$  são:

$$\alpha_i = \frac{x_{k-n_i} - x_{k-1-n_i}}{T_s}; \quad (4-18)$$

$$\beta_i = \frac{x_{k-n_i}}{T_s} (T_s - c_i) + \frac{x_{k-1-n_i}}{T_s} c_i.$$

Observa-se que, ao variar  $\mu$  nesse intervalo, o estado varia de aproximadamente  $x(kT_s - \tau_i)$  a exatamente  $x((k-n_i)T_s)$ . A primeira das parcelas da soma em (4-15) é aproximadamente calculada como a seguir:

$$\begin{aligned}
& e^{A_c T_s} \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} x(\mu + (k-n_i)T_s - c_i) d\mu \right] \cong \\
& \cong e^{A_c T_s} \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} \alpha_i \mu d\mu + \int_0^{c_i} e^{-A_c \mu} A_{c_i} \beta_i d\mu \right] = \\
& = e^{A_c T_s} \left\{ \left[ \int_0^{c_i} e^{-A_c \mu} \mu d\mu \right] A_{c_i} \alpha_i + \left[ \int_0^{c_i} e^{-A_c \mu} d\mu \right] A_{c_i} \beta_i \right\}.
\end{aligned} \tag{4-19}$$

Substituindo (4-18) em (4-19):

$$\begin{aligned}
& \frac{e^{A_c T_s}}{T_s} \left\{ \left[ \int_0^{c_i} e^{-A_c \mu} \mu d\mu \right] A_{c_i} \alpha_i + \left[ \int_0^{c_i} e^{-A_c \mu} d\mu \right] A_{c_i} \beta_i \right\} = \\
& = \frac{e^{A_c T_s}}{T_s} \left\{ \left[ \int_0^{c_i} e^{-A_c \mu} \mu d\mu \right] A_{c_i} (x_{k-n_i} - x_{k-1-n_i}) + \left[ \int_0^{c_i} e^{-A_c \mu} d\mu \right] A_{c_i} x_{k-n_i} (T_s - c_i) + \right. \\
& \quad \left. + \left[ \int_0^{c_i} e^{-A_c \mu} d\mu \right] A_{c_i} x_{k-1-n_i} c_i \right\}.
\end{aligned} \tag{4-20}$$

Vale lembrar que  $x_k = x(kT_s)$ , para qualquer  $k$ .

#### 4.2.2 De $c_i$ a $T_s$

De acordo com a Figura 4-2, no intervalo de  $c_i$  a  $T_s$ , os valores de  $\alpha$  e  $\beta$  são:

$$\begin{aligned}
\alpha_i &= \frac{x_{k+1-n_i} - x_{k-n_i}}{T_s}; \\
\beta_i &= \frac{x_{k-n_i}}{T_s} (T_s + c_i) - \frac{x_{k+1-n_i}}{T_s} c_i.
\end{aligned} \tag{4-21}$$

Observa-se que, ao variar  $\mu$  nesse intervalo, o estado varia de exatamente  $x((k-n_i)T_s)$  a aproximadamente  $x((k+1)T_s - \tau_i)$ . A segunda parcela da soma em em (4-15) é aproximadamente calculada como:

$$\begin{aligned}
& e^{A_c T_s} \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} x(\mu + (k - n_i) T_s - c_i) d\mu \right] \cong \\
& \cong e^{A_c T_s} \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} \alpha_i \mu d\mu + \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} \beta_i d\mu \right] = \\
& = \frac{e^{A_c T_s}}{T_s} \left\{ \left[ \int_{c_i}^{T_s} e^{-A_c \mu} \mu d\mu \right] A_{c_i} (x_{k+1-n_i} - x_{k-n_i}) + \left[ \int_{c_i}^{T_s} e^{-A_c \mu} d\mu \right] A_{c_i} x_{k-n_i} (T_s + c_i) - \right. \\
& \quad \left. - \left[ \int_{c_i}^{T_s} e^{-A_c \mu} d\mu \right] A_{c_i} x_{k+1-n_i} c_i \right\}. \tag{4-22}
\end{aligned}$$

### 4.2.3 Determinação dos termos $^{pos}A_{di}$ , $^{act}A_{di}$ , e $^{pre}A_{di}$

As matrizes  $^{pos}A_{di}$ ,  $^{act}A_{di}$ , e  $^{pre}A_{di}$  estão associadas aos estados  $x_{k+1-n_i}$ ,  $x_{k-n_i}$  e  $x_{k-1-n_i}$ , respectivamente. Portanto, são calculadas a partir das equações (4-20) e (4-22), resultando:

$$^{pos}A_{di} = \frac{e^{A_c T_s}}{T_s} \left\{ - \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} \mu d\mu \right] + \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} d\mu \right] c_i \right\}; \tag{4-23a}$$

$$\begin{aligned}
^{act}A_{di} = \frac{e^{A_c T_s}}{T_s} \left\{ + \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} \mu d\mu \right] + \left[ \int_0^{c_i} e^{-A_c \mu} A_{c_i} d\mu \right] (T_s - c_i) - \right. \\
\left. - \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} \mu d\mu \right] + \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} d\mu \right] (T_s + c_i) \right\}; \tag{4-23b}
\end{aligned}$$

$$^{pre}A_{di} = \frac{e^{A_c T_s}}{T_s} \left\{ + \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} \mu d\mu \right] - \left[ \int_{c_i}^{T_s} e^{-A_c \mu} A_{c_i} d\mu \right] c_i \right\}, \tag{4-23c}$$

lembrando que, a partir da equação (4-14):

$$\int_0^{T_s} e^{-A_c(\mu-T_s)} A_{c_i} x(\mu + kT_s - \tau_i) d\mu \cong ^{pos}A_{di} x_{k+1-n_i} + ^{act}A_{di} x_{k-n_i} + ^{pre}A_{di} x_{k-1-n_i}. \tag{4-24}$$

O apêndice A mostra as fórmulas em que se baseiam os algoritmos destinados a calcular as integrais em (4-23).

Desse modo, um sistema de tempo contínuo com atrasos de entrada e realimentação de estados pode ser aproximadamente representado por um sistema de tempo discreto. Quando os atrasos não são múltiplos do período de amostragem, a solução apresentada acima é a interpolação dos pontos para solucionar a integral nos intervalos considerados.

### 4.3 A Saída

As matrizes  ${}^{act}G_{di}$  e  ${}^{pre}G_{di}$  são formadas pela interpolação de estados para gerar a saída desejada a partir de  $G_{ci}$ .

A partir da Figura 4-2, faz-se uma aproximação linear entre dois estados consecutivos para obter o estado localizado entre esses dois períodos de amostragem. Assim, dado um estado qualquer  $x(t-\tau_i)$ , esse estado pode ser aproximado por:

$$x(t-\tau_i) \cong \frac{x_{k-1-n_i} - x_{k-n_i}}{T_s} c_i + x_{k-n_i}. \quad (4-25)$$

Dessa forma, determinam-se as matrizes de ganhos para as saídas defasadas:

$$G_{ci}x(t-\tau_i) \cong G_{ci} \left[ \frac{x_{k-1-n_i} - x_{k-n_i}}{T_s} c_i + x_{k-n_i} \right]; \quad (4-26)$$

$$G_{ci}x(t-\tau_i) \cong G_{ci} \left[ 1 - \frac{c_i}{T_s} \right] x_{k-n_i} + G_{ci} \frac{c_i}{T_s} x_{k-1-n_i}.$$

Portanto:

$${}^{act}G_{di} \cong C_{ci} \left[ 1 - \frac{c_i}{T_s} \right]; \quad (4-27a)$$

$${}^{pre}G_{di} \cong C_{ci} \frac{c_i}{T_s}, \quad (4-27b)$$

são as matrizes das saídas, de acordo com o modelo discreto proposto em (4-4). A linearidade dessas transformações, torna o resultado em (4-27) válido para qualquer valor de  $i$ .

### 4.4 Modelo de Tempo Discreto e Regime Permanente

Uma aspecto que pode ser analisado a partir do modelo de tempo discreto é o estado síncrono da rede. Para tanto, precisa-se obter o estado de regime permanente para que se possa extrair a frequência de sincronismo para comparar com a fornecida pela equação (3-8), a qual é baseada no modelo de tempo contínuo. Para tanto, convém obter uma realização da equação (4-4) com a forma:

$$\begin{aligned} z_{k+1} &= A_z z_k + B_z u_k; \\ y_k &= G_z z_k. \end{aligned} \quad (4-28)$$

o que é factível, visto que se tem um sistema de tempo discreto linear. O apêndice B fornece a listagem do programa em MATLAB chamado DISCRETIZA para gerar o modelo de tempo discreto. Detalhando um pouco mais o sistema em (4-28), suas constantes e estados podem ser descritos como:

$$z_k = \begin{bmatrix} x_{k-n} \\ x_{k-n+1} \\ x_{k-n+2} \\ \vdots \\ x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix}; \quad A_z = \begin{bmatrix} 0 & I & 0 & \dots & 0 & 0 & 0 \\ 0 & 0 & I & \dots & 0 & 0 & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 0 & I & 0 \\ 0 & 0 & 0 & \dots & 0 & 0 & I \\ {}^n A_d & {}^{n-1} A_d & {}^{n-2} A_d & \dots & {}^2 A_d & {}^1 A_d & {}^0 A_d \end{bmatrix}; \quad B_z = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ {}^0 B_d \end{bmatrix}, \quad (4-29)$$

sendo  ${}^h A_d$  e  ${}^0 B_d$ ,  $h=0,1,2,\dots,n$ , as matrizes obtidas a partir dos ganhos do modelo (4-4) e  $I$  a matriz identidade. O estado de tempo discreto  $x_{k-h}$  é equivalente ao estado de tempo contínuo  $x(t)$  e é composto pelas fases e estados dos filtros: A equação (4-30) mostra um exemplo:

$$x_{k-h} = \begin{bmatrix} \theta_{1\ k-h} & x_{1\ k-h} & \dot{x}_{1\ k-h} & \theta_{2\ k-h} & \dots & \theta_{j\ k-h} & x_{j\ k-h} & \dot{x}_{j\ k-h} & \dots \end{bmatrix}. \quad (4-30)$$

Vide também a expressão (3-18) para a rede de três nós com filtros de segunda ordem.

Para determinar o estado síncrono, considera-se que os valores das fases crescem à uma taxa constante  $\omega_s$ . Para um nó  $j$  qualquer, o valor de sua fase fica:

$$\theta_j(t) = \omega_s t + \varphi_j, \quad (4-31)$$

onde  $\varphi_j$  é uma constante. Passando (4-31) para o domínio discreto, com período de amostragem  $T_s$ , temos:

$$\theta_{j\ k+1} = \theta_{j\ k} + T_s \omega_s, \quad (4-32)$$

sendo a condição inicial  $\theta_{j\ 0} = \varphi_j$ . A equação (4-32) é uma rampa discreta com declividade  $\omega_s$ . Modificando (4-32), a fase do nó  $j$  que compõe o estado  $x_{k-h}$  é:

$$\theta_{j\ k-h+1} = \theta_{j\ k-h} + T_s \omega_s. \quad (4-33)$$

No estado síncrono, supondo, a princípio, que o valor da fase do DPLL 1 em  $x_{k-n} - \theta_{1\ k-n}$  seja 0 em um dado instante, tem-se  $\theta_{1\ k-n+1} = \omega_s \cdot T_s$ , ou, como  $\theta_{1\ k-n+1} = \omega_s \cdot T_s$ :

$$\theta_{j\ k-h+1} = \theta_{j\ k-h} + \theta_{1\ k-n+1}. \quad (4-34)$$

Por pertencer ao estado  $z_k$ , o qual se deseja determinar, não será imposto explicitamente que  $\theta_{1\ k-n}$  seja igual a 0. Porém, seu efeito será anulado na equação (4-28), tornando a primeira coluna de  $A_z$  um vetor nulo. Feito disso, reescreve-se  $z_{k+1}$  em função de  $z_k$ , somando o termo  $\omega_s \cdot T_s$  – via  $\theta_{1\ k-n+1}$  – às fases  $\theta_{j\ k-h}$ , de acordo com a equação (4-34). Para tanto, define-se a matriz  $D$ , de modo que (4-28) fica, no regime permanente:



$$D \begin{bmatrix} x_{k-n} \\ x_{k-n+1} \\ x_{k-n+2} \\ \vdots \\ x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix} = \begin{bmatrix} 0 & I & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & I & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & I & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & I \\ {}^n A_d' & {}^{n-1} A_d & {}^{n-2} A_d & \cdots & {}^2 A_d & {}^1 A_d & {}^0 A_d \end{bmatrix} \begin{bmatrix} x_{k-n} \\ x_{k-n+1} \\ x_{k-n+2} \\ \vdots \\ x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ {}^1 B_d \end{bmatrix} u_k, \quad (4-35)$$

sendo  ${}^n A_d'$  a matriz  ${}^n A_d$  com a primeira coluna nula.  $D$  acrescenta o termo  $\omega_s \cdot T_s$  – como se este fosse um estado que compõe  $z_k$  – a cada fase, em todos os nós  $j$  que fazem parte dos estados  $x_{k-h}$ . Os valores da diagonal principal de  $D$ , bem como as linhas de sua primeira coluna que correspondem às fases que compõem o estado  $z_k$ , são iguais a 1. Um exemplo de como fica essa matriz, para filtros de segunda ordem, é dado por:

$$D = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & \cdots & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & \cdots & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (4-36)$$

O estado de regime fica, portanto:

$$\begin{bmatrix} x_{k-n} \\ x_{k-n+1} \\ x_{k-n+2} \\ \vdots \\ x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix} = D \left( \begin{bmatrix} 0 & I & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & I & \cdots & 0 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 0 & I & 0 \\ 0 & 0 & 0 & \cdots & 0 & 0 & I \\ {}^n A_d' & {}^{n-1} A_d & {}^{n-2} A_d & \cdots & {}^2 A_d & {}^1 A_d & {}^0 A_d \end{bmatrix} \right)^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ {}^0 B_d \end{bmatrix} u_k. \quad (4-37)$$

Assim, obtém-se  $\omega_s$  a partir de  $\theta_{l \ k-n}$  do modelo de tempo discreto – dado que o período de amostragem  $T_s$  é conhecido – bem como as saídas dos filtros ou as diferenças de fase estacionárias entre os DPLLs, pois as mesmas são constantes em regime.

No capítulo a seguir, são mostrados alguns resultados obtidos para diferentes configurações de malhas. Baseados em seus parâmetros, serão construídos os modelos de tempo discreto e, a partir deles, serão determinados seus estados síncronos e a estabilidade dos mesmos será avaliada.

## 4.5 Modelo de Tempo Discreto e Estabilidade<sup>8</sup>

Uma vez encerrado o processo de discretização e determinado o estado síncrono, inicia-se a análise da estabilidade de uma rede mutuamente conectada de DPLLs. Fazendo uma breve recapitulação, obtém-se primeiramente o modelo no tempo contínuo do DPLL, o qual forma cada um dos nós da rede. Modela-se esta rede no espaço de estados, a qual é, finalmente, discretizada, encontrando-se todos os elementos do sistema em (4-4).

Para analisar a estabilidade de (4-4), obtém-se uma realização na forma da equação (4-28) da malha de DPLLs, passando pela determinação dos autovalores da matriz  $A_z$ . A rede é estável se todos esses autovalores estão contidos no interior do disco de raio unitário e de centro na origem do plano complexo. Isto é, para todo autovalor  $\rho$  de  $A_z$ :

$$|\rho| \leq 1. \quad (4-38)$$

Caso contrário, o sistema é instável. Portanto, dada uma malha de sincronismo estável, os modelos de tempo discreto construídos a partir dos parâmetros dessa malha, com a forma da equação (4-28), devem também ser estáveis, de modo que o valor máximo dentre os módulos dos autovalores de  $A_z$  seja menor ou igual a 1. Isto é:

$$\max_{\rho \in \Lambda(A_z)} (|\rho|) \leq 1, \quad (4-39)$$

sendo  $\Lambda(A_z)$  o conjunto dos autovalores de  $A_z$ .

No capítulo 0, isso será verificado com a construção de modelos de tempo discreto baseados nos parâmetros das redes de sincronismo. De posse desses modelos, poder-se-á determinar o estado síncrono e avaliar a estabilidade do modelo. Espera-se que o comportamento do modelo real da rede seja semelhante ao equivalente de tempo discreto.

---

<sup>8</sup> ([24])

## Capítulo 5

### Resultados e Simulações

A seguir são mostrados alguns resultados para redes de quatro e cinco nós, divididos em casos distintos. Em cada caso, os parâmetros dos filtros são fornecidos e simulações em MATLAB dos sistemas de tempo contínuo e seus equivalentes de tempo discreto são apresentadas. As simulações do modelo real também são apresentadas. Chama-se *modelo real* – ou *sistema real* – aquele cujos algoritmos matemáticos representam mais fielmente os dispositivos reais, inclusive as não-linearidades. Dos modelos de tempo contínuo e discreto serão obtidas as frequências sincronismo. E a partir dos modelos de tempo discreto serão analisadas a estabilidade dos sistemas.

As condições iniciais adotadas estão próximas do estado estacionário. As frequências de excitação estão próximas entre si, com no máximo 10Hz. Os atrasos entre os nós são aleatórios e estão compreendidos entre 0,01s e 0,10s. Os períodos de amostragem também são aleatórios, com no mínimo 0,01s e no máximo 1,00s. Os nós são de segunda ou terceira ordens, como mencionado no item 2.4.1. As constantes do filtro e do detector de fases foram normalizadas, com  $K_f = 1$  e  $K_d = 1/4\pi$ , sendo que a saída do detector pode variar de  $-0.5$  a  $0.5$ . A resposta dos filtros foram escolhidas para compor as saídas  $y$  dos modelos, pois é através delas que poderemos observar o comportamento das malhas.

## 5.1 Redes com Quatro Nós

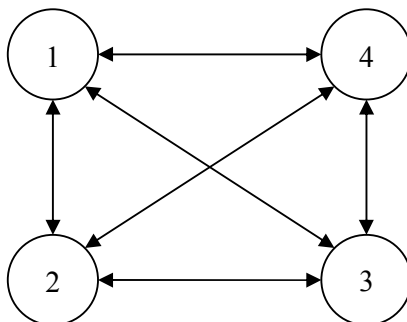


Figura 5-1: Rede com quatro nós mutuamente conectados.

A Figura 5-1 ilustra a rede com quatro nós. Esta rede possui seis conexões cujos atrasos iremos definir como:

$$\left\{ \begin{array}{l} \tau_{12} = \tau_1; \\ \tau_{13} = \tau_2; \\ \tau_{14} = \tau_3; \end{array} \right. \left\{ \begin{array}{l} \tau_{23} = \tau_4; \\ \tau_{24} = \tau_5; \\ \tau_{34} = \tau_6. \end{array} \right. \quad (5-1)$$

### 5.1.1 Caso 1

Tabela 5-1 – Resultados para rede de quatro nós – caso 1			
Frequência Central [Hz]	Ganho [Hz/V]	Filtro [V/rad]	Amostragem [s]
$f_1 = 6,5493$	$K_{o1} = 13,0986$	$\frac{1}{s+1}$	$T_s = 0,067$
$f_2 = 6,6344$	$K_{o2} = 13,2687$		
$f_3 = 6,7194$	$K_{o3} = 13,4389$		
$f_4 = 6,8045$	$K_{o4} = 13,6090$		
Atrasos [s]			
$\tau_1 = 0,0114$	$\tau_2 = 0,0772$	$\tau_3 = 0,0501$	$\tau_4 = 0,0939$
$\tau_5 = 0,0519$	$\tau_6 = 0,0477$		
Respostas			
Frequência de sincronismo [Hz]		Autovalor Máximo Discreto	Saída dos Filtros [V]
Contínuo	Discreto		
4,8744	4,8744	1,0000	$u_{f1} = -0,1279$
			$u_{f2} = -0,1326$
			$u_{f3} = -0,1373$
			$u_{f4} = -0,1418$

- *Composição do estado*

Aqui o estado é formado por quatro valores de fases e pelos estados dos filtros. Como o filtro é de primeira ordem, temos quatro variáveis compondo o estado completo. Este estado será utilizado tanto para formar o modelo de tempo contínuo –  $x(t)$  – e o de tempo discreto –  $x_{k-h}$ , com  $h = 1, 2, \dots$ .

$$x = [\theta_1 \quad v_{f1} \quad \theta_2 \quad v_{f2} \quad \theta_3 \quad v_{f3} \quad \theta_4 \quad v_{f4}]^T. \quad (5-2)$$

- *Sistema de tempo contínuo*

O sistema de tempo contínuo possui seis atrasos distintos, dado que temos quatro nós. E os ganhos  $A_{c1}$  a  $A_{c6}$  representam os ganhos dos estados atrasados.

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t) + A_{c1} x(t - \tau_1) + A_{c2} x(t - \tau_2) + A_{c3} x(t - \tau_3) + \\ &\quad A_{c4} x(t - \tau_4) + A_{c5} x(t - \tau_5) + A_{c6} x(t - \tau_6); \\ y(t) &= G_c x(t). \end{aligned} \quad (5-3)$$

Abaixo temos cada uma das matrizes que constituem o sistema (5-3). Vale notar que as matrizes foram desmembradas devido ao grande número de elementos nulos. As matrizes nulas são representados por  $0_{r \times p}$ , sendo aqui  $r$  o número de linhas e  $p$  o número de colunas.

$$A_c = \begin{bmatrix} {}^1A_c & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & {}^2A_c & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & {}^3A_c & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & {}^4A_c \end{bmatrix}; \quad (5-4a)$$

$${}^1A_c = \begin{bmatrix} 0 & 82,3011 \\ -0,0796 & -1 \end{bmatrix}; \quad {}^2A_c = \begin{bmatrix} 0 & 83,3699 \\ -0,0796 & -1 \end{bmatrix}; \quad (5-4b)$$

$${}^3A_c = \begin{bmatrix} 0 & 84,4388 \\ -0,0796 & -1 \end{bmatrix}; \quad {}^4A_c = \begin{bmatrix} 0 & 85,5076 \\ -0,0796 & -1 \end{bmatrix}; \quad (5-4c)$$

$$B_c = \begin{bmatrix} 6,2832 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 6,2832 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 6,2832 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6,2832 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \quad (5-4d)$$

$$A_{c1} = \begin{bmatrix} 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} \\ {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}; \quad A_{c2} = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}; \quad (5-4e)$$

$$A_{c3} = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}; \quad A_{c4} = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} \\ 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}; \quad (5-4f)$$

$$A_{c5} = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} & 0_{2 \times 2} \end{bmatrix}; \quad A_{c6} = \begin{bmatrix} 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} \\ 0_{2 \times 2} & 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c \\ 0_{2 \times 2} & 0_{2 \times 2} & {}^0A_c & 0_{2 \times 2} \end{bmatrix}; \quad (5-4g)$$

$${}^0A_c = \begin{bmatrix} 0 & 0 \\ 0,0265 & 0 \end{bmatrix}; \quad (5-4h)$$

$$G_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (5-4i)$$

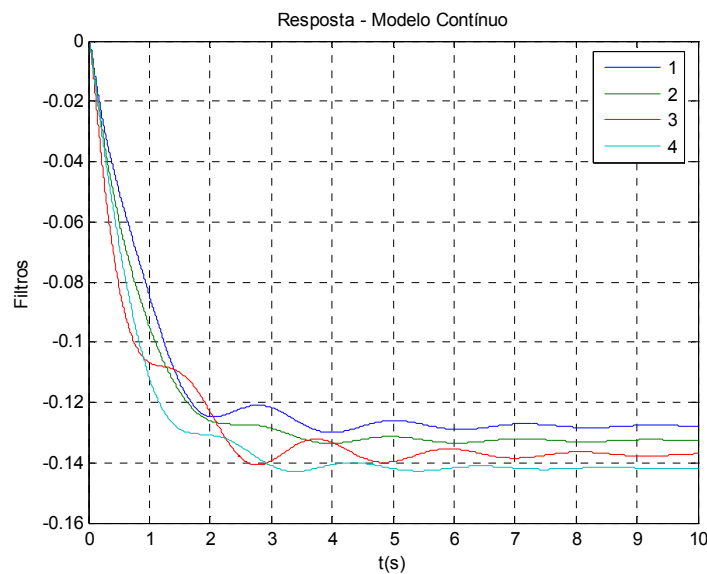


Figura 5-2: Malha com quatro nós: resposta do modelo de tempo contínuo do caso 1.

A Figura 5-2 mostra a saída ponderada dos filtros do modelo de tempo contínuo. Isso quer dizer que as respostas dos filtros levam em conta a ponderação de fases. Assim, há quatro linhas distintas no gráfico que se referem a cada um dos quatro nós.

- *Sistema de tempo discreto*

As equações a seguir mostram o resultado da discretização do sistema (5-3). A dimensão do sistema discretizado depende fundamentalmente do período de amostragem adotado e dos valores dos atrasos.

$$\begin{bmatrix} x_{k-1} \\ x_k \\ x_{k+1} \end{bmatrix} = \begin{bmatrix} 0_{8 \times 8} & I & 0_{8 \times 8} \\ 0_{8 \times 8} & 0_{8 \times 8} & I \\ {}^2A_d & {}^1A_d & {}^0A_d \end{bmatrix} \begin{bmatrix} x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix} + \begin{bmatrix} 0_{16 \times 8} \\ 0 \\ B_d \end{bmatrix} u_k; \quad (5-5)$$

$$y_k = \begin{bmatrix} 0_{8 \times 16} & {}^0G_d \end{bmatrix}.$$

As matrizes a seguir são o detalhamento do sistema (5-5).

$${}^0A_d = \begin{bmatrix} 0,9857 & 5,3074 & 0,0046 & 0,0050 & 0,0010 & 0,0000 & 0,0028 & 0,0001 \\ -0,0051 & 0,9212 & 0,0017 & 0,0032 & 0,0006 & 0,0000 & 0,0012 & 0,0003 \\ 0,0047 & 0,0050 & 0,9855 & 5,3760 & 0,0004 & 0,0000 & 0,0027 & 0,0001 \\ 0,0017 & 0,0032 & -0,0051 & 0,9210 & 0,0003 & 0,0000 & 0,0012 & 0,0002 \\ 0,0010 & 0,0000 & 0,0004 & 0,0000 & 0,9853 & 5,4446 & 0,0030 & 0,0002 \\ 0,0006 & 0,0000 & 0,0003 & 0,0000 & -0,0051 & 0,9208 & 0,0013 & 0,0004 \\ 0,0029 & 0,0001 & 0,0028 & 0,0001 & 0,0031 & 0,0002 & 0,9851 & 5,5131 \\ 0,0012 & 0,0003 & 0,0012 & 0,0002 & 0,0013 & 0,0004 & -0,0051 & 0,9206 \end{bmatrix}; \quad (5-6a)$$

$${}^1A_d = \begin{bmatrix} 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0037 & 0,0000 & 0,0020 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0011 & 0,0000 & 0,0005 & 0,0000 \\ 0,0001 & 0,0000 & 0,0000 & 0,0000 & 0,0038 & 0,0000 & 0,0021 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0013 & 0,0000 & 0,0005 & 0,0000 \\ 0,0038 & 0,0000 & 0,0039 & 0,0000 & 0,0000 & 0,0000 & 0,0019 & 0,0000 \\ 0,0011 & 0,0000 & 0,0013 & 0,0000 & 0,0000 & 0,0000 & 0,0004 & 0,0000 \\ 0,0021 & 0,0000 & 0,0022 & 0,0000 & 0,0019 & 0,0000 & 0,0000 & 0,0000 \\ 0,0005 & 0,0000 & 0,0005 & 0,0000 & 0,0004 & 0,0000 & 0,0000 & 0,0000 \end{bmatrix}; \quad (5-6b)$$

$${}^2A_d = \begin{bmatrix} 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0007 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0000 \\ 0,0001 & 0,0000 & 0,0007 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \end{bmatrix}; \quad (5-6c)$$

$${}^0B_d = \begin{bmatrix} 0,4189 & 0,0004 & 0,0000 & 0,0000 \\ -0,0011 & 0,0003 & 0,0000 & 0,0000 \\ 0,0004 & 0,4189 & 0,0000 & 0,0000 \\ 0,0003 & -0,0011 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,4189 & 0,0000 \\ 0,0000 & 0,0000 & -0,0011 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,4189 \\ 0,0000 & 0,0000 & 0,0000 & -0,0011 \end{bmatrix}; \quad (5-6d)$$

$${}^0G_d = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (5-6e)$$

Os autovalores em (5-7) foram obtidos a partir do sistema (5-5), o qual é estável, sendo que o maior dos módulos dos autovalores é menor ou igual a 1, como mostrado na Tabela 5-1.

Autovalores :

$$\left\{ \begin{array}{l} \rho_1 = 0; \\ \rho_2 = 0; \\ \rho_3 = 0; \\ \rho_4 = 0; \\ \rho_5 = 0; \\ \rho_6 = 0; \end{array} \right\} \left\{ \begin{array}{l} \rho_7 = 0; \\ \rho_8 = 0; \\ \rho_9 = 0; \\ \rho_{10} = 1; \\ \rho_{11} = 0,9109; \\ \rho_{12} = 0,9567 + 0,1910i; \end{array} \right\} \left\{ \begin{array}{l} \rho_{13} = 0,9567 - 0,1910i; \\ \rho_{14} = 0,9509 + 0,1915i; \\ \rho_{15} = 0,9509 - 0,1915i; \\ \rho_{16} = 0,9494 + 0,1883i; \\ \rho_{17} = 0,9494 - 0,1883i; \\ \rho_{18} = 0,0133; \end{array} \right\} \left\{ \begin{array}{l} \rho_{19} = -0,0023 + 0,0100i; \\ \rho_{20} = -0,0023 - 0,0100i; \\ \rho_{21} = -0,0086; \\ \rho_{22} = 0,0007; \\ \rho_{23} = -0,0007; \\ \rho_{24} = 0,0000. \end{array} \right. \quad (5-7)$$

A Figura 5-3 mostra a resposta ponderada dos filtros do sistema de tempo discreto. A largura de cada degrau equivale ao período de amostragem e o fato de o mesmo ter sido escolhido pequeno, contribuiu para a precisão do modelo de tempo discreto.

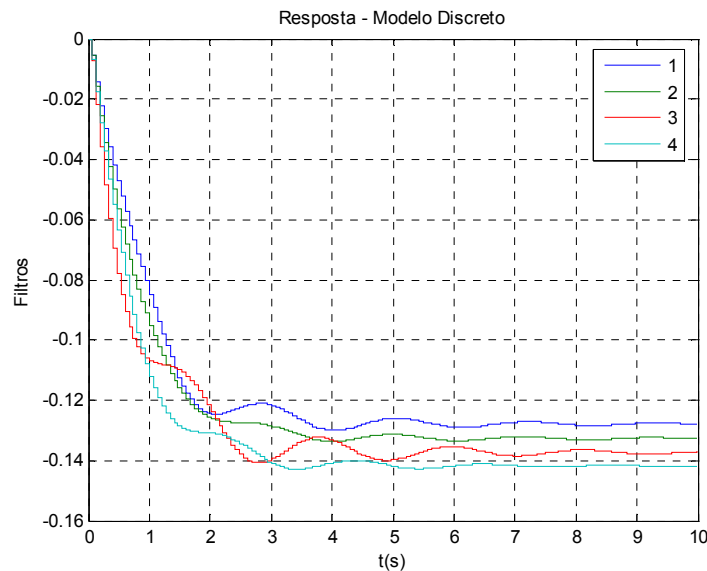


Figura 5-3: Malha com quatro nós: resposta do modelo de tempo discreto do caso 1.

O erro mostrado na Figura 5-4 deixa evidente a precisão do modelo discretizado em relação ao seu equivalente de tempo contínuo. Este erro tende a ser nulo à medida que o tempo tende a infinito, dado que os estados síncronos determinados para ambos os modelos – como se pode notar pelas frequências de sincronismo – são idênticos.



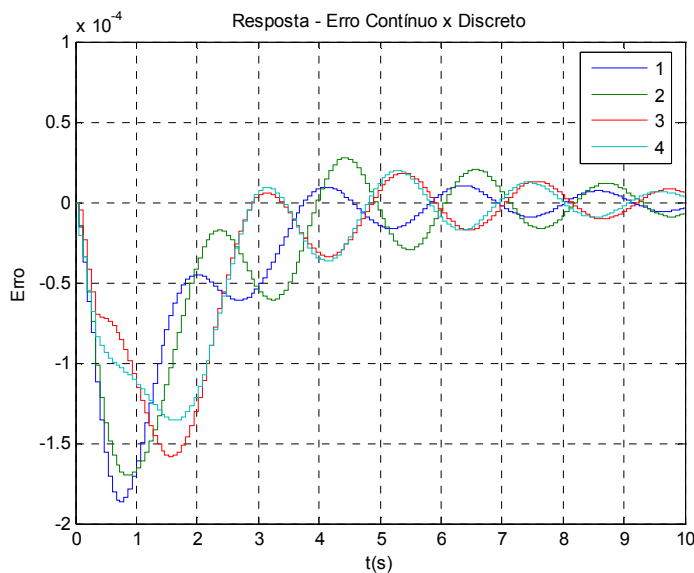


Figura 5-4: Malha com quatro nós: resposta do erro contínuo  $\times$  discreto do caso 1.

- *Sistema real*

As duas próximas figuras – Figura 5-5 e Figura 5-6 – mostram a resposta da simulação para o sistema real.

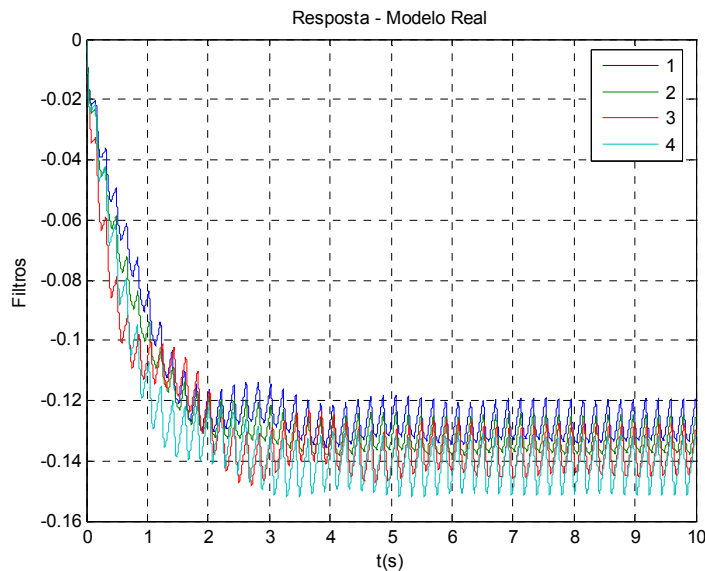


Figura 5-5: Malha com quatro nós: resposta do sistema real do caso 1.

Esses modelos foram construídos de modo que se comportassem como uma malha de DPLLs reais. Os modelos lineares aproximados não foram utilizados aqui. Portanto, os sinais gerados

pelos VCOs possuem formas de onda quadradas, as quais são diretamente aplicadas aos detectores de fase. Esses últimos foram dimensionados para responder às bordas dos sinais aplicados às suas entradas, seguindo o comportamento do comparador “charge-pump”, descrito no Capítulo 2. Os filtros foram implementados de forma a eliminar a influência do estado “tri-state” do detector de fases, similar ao circuito da Figura 2-9, também no Capítulo 2.

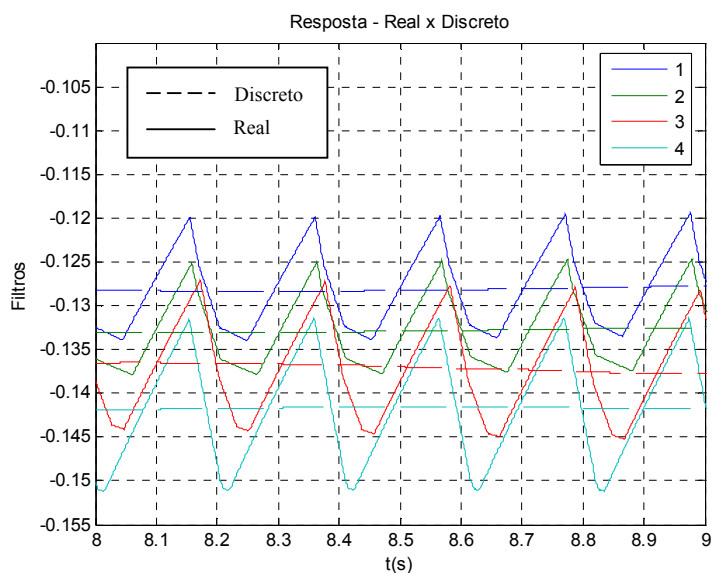


Figura 5-6: Malha com quatro nós: resposta real  $\times$  discreto do caso 1.

A Figura 5-6 mostra as respostas tanto do sistema real como do discretizado, ambos próximos do regime. Os valores médios dos sinais reais estão próximos dos valores obtidos a partir do sistema discretizado. A forma *dente-de-serra* na resposta dos filtros do sistema real se deve à forma de onda quadrada gerada na saída do detector que, ao passar pelo filtro, tem suas harmônicas de maior ordem atenuadas. Este fenômeno é equivalente às consecutivas cargas e descargas de um capacitor.

## 5.1.2 Caso 2

Tabela 5-2 – Resultados para rede de quatro nós – caso 2			
Frequência Central [Hz]	Ganho [Hz/V]	Filtro [V/rad]	Amostragem [s]
$f_1 = 9,2888$	$K_{o1} = 18,5775$	$\frac{s + 0,375}{s^2 + 0,750s + 0,125}$	$T_s = 0,13$
$f_2 = 9,4094$	$K_{o2} = 18,8188$		
$f_3 = 9,5300$	$K_{o3} = 19,0601$		
$f_4 = 9,6507$	$K_{o4} = 19,3013$		
Atrasos [s]			
$\tau_1 = 0,0955$	$\tau_2 = 0,0308$	$\tau_3 = 0,0646$	$\tau_4 = 0,0537$
$\tau_5 = 0,0902$	$\tau_6 = 0,0786$		
Respostas			
Frequência de sincronismo [Hz]		Autovalor Máximo Discreto	Saída dos Filtros em Regime [V]
Contínuo	Discreto		
3,2015	3,2015	1,0000	$u_{f1} = -0,3277$
			$u_{f2} = -0,3399$
			$u_{f3} = -0,3320$
			$u_{f4} = -0,3341$

- *Composição do estado*

Aqui, como se pode notar, a dimensão do estado é maior que no caso 1, pois o filtro utilizado é de segunda ordem. Portanto, temos um estado com doze variáveis.

$$x = [\theta_1 \quad v_{f1} \quad \dot{v}_{f1} \quad \theta_2 \quad v_{f2} \quad \dot{v}_{f2} \quad \theta_3 \quad v_{f3} \quad \dot{v}_{f3} \quad \theta_4 \quad v_{f4} \quad \dot{v}_{f4}]^T. \quad (5-8)$$

- *Sistema de tempo contínuo*

A descrição do sistema de tempo contínuo em nada difere do primeiro caso. Ainda tem-se seis estados atrasados. Porém, a dimensão do sistema torna-se maior na medida que a ordem do filtro é maior.

$$\begin{aligned} \dot{x}(t) = & A_c x(t) + B_c u(t) + A_{c1} x(t - \tau_1) + A_{c2} x(t - \tau_2) + A_{c3} x(t - \tau_3) + \\ & A_{c4} x(t - \tau_4) + A_{c5} x(t - \tau_5) + A_{c6} x(t - \tau_6); \end{aligned} \quad (5-9)$$

$$y(t) = G_c x(t).$$

Cada matriz de (5-9) está descrita a seguir.

$$A_c = \begin{bmatrix} {}^1A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^2A_c & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^3A_c & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^4A_c \end{bmatrix}, \quad (5-10a)$$

$${}^1A_c = \begin{bmatrix} 0 & 43,7723 & 116,7262 \\ 0 & 0 & 1 \\ -0,0796 & -0,1250 & -0,7500 \end{bmatrix}; \quad {}^2A_c = \begin{bmatrix} 0 & 44,3408 & 118,2421 \\ 0 & 0 & 1 \\ -0,0796 & -0,1250 & -0,7500 \end{bmatrix}; \quad (5-10b)$$

$${}^3A_c = \begin{bmatrix} 0 & 44,9093 & 119,7850 \\ 0 & 0 & 1 \\ -0,0796 & -0,1250 & -0,7500 \end{bmatrix}; \quad {}^4A_c = \begin{bmatrix} 0 & 45,4777 & 121,2739 \\ 0 & 0 & 1 \\ -0,0796 & -0,1250 & -0,7500 \end{bmatrix}; \quad (5-10c)$$

$$B_c = \begin{bmatrix} 6,2832 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 6,2832 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 6,2832 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 6,2832 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}; \quad (5-10d)$$

$$A_{c1} = \begin{bmatrix} 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \\ {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c2} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad (5-10e)$$

$$A_{c3} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c4} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad (5-10f)$$

$$A_{c5} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c6} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \end{bmatrix}; \quad (5-10g)$$

$${}^0A_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0,0265 & 0 & 0 \end{bmatrix}; \quad (5-10h)$$

$$G_c = \begin{bmatrix} 0 & 0,3750 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,3750 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0,3750 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,3750 & 1 \end{bmatrix}; \quad (5-10i)$$

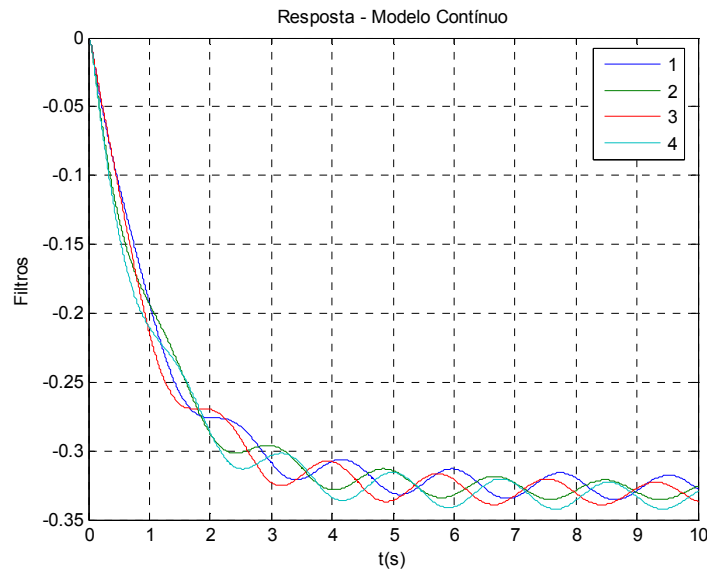


Figura 5-7: Malha com quatro nós: resposta do modelo de tempo contínuo do caso 2.

A resposta do modelo de tempo contínuo – com visto na Figura 5-7 – mostra um comportamento semelhante ao primeiro caso. Em ambos, a frequência de sincronismo é menor que as frequências naturais dos nós. Porém, aqui, como a soma dos atrasos entre os nós é maior que no primeiro caso, a frequência síncrona tende a ser menor no segundo, supondo idênticos os demais parâmetros. Vide equação (3-8).

- *Sistema de tempo discreto*

De acordo com a equação (5-11), o estado é composto apenas por  $x_k$  e  $x_{k-1}$ . Isso ocorre devido ao valor do período de amostragem ser maior que os valores de todos os atrasos. Quanto maior o valor do período em relação aos atrasos, menor tende a ser a dimensão do modelo discretizado.

$$\begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix} = \begin{bmatrix} 0_{1 \times 12} & I \\ {}^1A_d & 0_{A_d} \end{bmatrix} \begin{bmatrix} x_{k-1} \\ x_k \end{bmatrix} + \begin{bmatrix} 0_{1 \times 12} \\ {}^0B_d \end{bmatrix} u_k; \quad (5-11)$$

$$y_k = \begin{bmatrix} 0_{1 \times 12} & {}^0G_d \end{bmatrix} \cdot$$

As matrizes a seguir detalham mais o modelo de tempo discreto.

$${}^0A_d = \begin{bmatrix} 0,9239 & 5,4248 & 14,4269 & 0,0152 & 0,0009 & 0,0025 & 0,0238 & 0,0213 & 0,0565 & 0,0202 & 0,0062 & 0,0165 \\ -0,0006 & 0,9978 & 0,1206 & 0,0001 & 0,0000 & 0,0000 & 0,0002 & 0,0002 & 0,0005 & 0,0002 & 0,0001 & 0,0001 \\ -0,0096 & -0,0432 & 0,8324 & 0,0024 & 0,0007 & 0,0018 & 0,0031 & 0,0054 & 0,0144 & 0,0028 & 0,0024 & 0,0064 \\ 0,0154 & 0,0009 & 0,0025 & 0,9228 & 5,4933 & 14,6090 & 0,0219 & 0,0098 & 0,0261 & 0,0163 & 0,0014 & 0,0038 \\ 0,0001 & 0,0000 & 0,0000 & -0,0006 & 0,9977 & 0,1205 & 0,0002 & 0,0001 & 0,0002 & 0,0001 & 0,0000 & 0,0000 \\ 0,0024 & 0,00007 & 0,0018 & -0,0096 & -0,0435 & 0,8314 & 0,0029 & 0,0032 & 0,0086 & 0,0025 & 0,0009 & 0,0024 \\ 0,0245 & 0,0213 & 0,0566 & 0,0222 & 0,0098 & 0,0261 & 0,09220 & 5,5619 & 14,7915 & 0,0186 & 0,0031 & 0,0083 \\ 0,0002 & 0,0002 & 0,0005 & 0,0002 & 0,0001 & 0,0002 & -0,0006 & 0,9977 & 0,1205 & 0,0002 & 0,0000 & 0,0001 \\ 0,0031 & 0,0053 & 0,0140 & 0,0029 & 0,0032 & 0,0085 & -0,0095 & -0,0439 & 0,8305 & 0,0027 & 0,0015 & 0,0040 \\ 0,0209 & 0,0062 & 0,0165 & 0,0167 & 0,0014 & 0,0038 & 0,0187 & 0,0031 & 0,0083 & 0,9209 & 5,6303 & 14,9733 \\ 0,0002 & 0,0001 & 0,0001 & 0,0001 & 0,0000 & 0,0000 & 0,0002 & 0,0000 & 0,0001 & -0,0006 & 0,9977 & 0,1204 \\ 0,0028 & 0,0023 & 0,0062 & 0,0025 & 0,0009 & 0,0023 & 0,0026 & 0,0015 & 0,0040 & -0,0096 & -0,0443 & 0,8295 \end{bmatrix} \quad (5-12a)$$

$${}^1A_d = \begin{bmatrix} 0,0000 & 0,0000 & 0,0000 & 0,0103 & 0,0000 & 0,0000 & 0,0013 & 0,0000 & 0,0000 & 0,0052 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0008 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0004 & 0,0000 & 0,0000 \\ 0,0104 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0037 & 0,0000 & 0,0000 & 0,0095 & 0,0000 & 0,0000 \\ 0,0001 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 \\ 0,0008 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 \\ 0,0013 & 0,0000 & 0,0000 & 0,0038 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0076 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 \\ 0,0001 & 0,0000 & 0,0000 & 0,0003 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0006 & 0,0000 & 0,0000 \\ 0,0054 & 0,0000 & 0,0000 & 0,0097 & 0,0000 & 0,0000 & 0,0076 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0000 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0001 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \\ 0,0004 & 0,0000 & 0,0000 & 0,0007 & 0,0000 & 0,0000 & 0,0006 & 0,0000 & 0,0000 & 0,0000 & 0,0000 & 0,0000 \end{bmatrix} \quad (5-12b)$$

$${}^0B_d = \begin{bmatrix} 0,7959 & 0,0001 & 0,0030 & 0,0009 \\ -0,0002 & 0,0000 & 0,0000 & 0,0000 \\ -0,0040 & 0,0001 & 0,00008 & 0,0003 \\ 0,0001 & 0,7956 & 0,0014 & 0,0002 \\ 0,0000 & -0,0002 & 0,0000 & 0,0000 \\ 0,0001 & -0,0040 & 0,0005 & 0,0001 \\ 0,0031 & 0,0014 & 0,7953 & 0,0004 \\ 0,0000 & 0,0000 & -0,0002 & 0,0000 \\ 0,0008 & 0,0005 & -0,0040 & 0,0002 \\ 0,0009 & 0,0002 & 0,0004 & 0,7951 \\ 0,0000 & 0,0000 & 0,0000 & -0,0002 \\ 0,0003 & 0,0001 & 0,0002 & -0,0040 \end{bmatrix} \quad (5-12c)$$

$${}^0G_d = \begin{bmatrix} 0 & 0,3750 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0,3750 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,3750 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0,3750 & 1 \end{bmatrix} \quad (5-12d)$$

Os autovalores em (5-13) indicam que o modelo de tempo discreto é estável, pois o maior dentre os módulos dos autovalores é 1, como indicado na Tabela 5-2. Portanto, espera-se que também a rede de sincronismo real seja estável próximo ao estado síncrono.

$$\text{Autovalores } s : \quad \left\{ \begin{array}{l} \rho_1 = 0; \\ \rho_2 = 0; \\ \rho_3 = 0; \\ \rho_4 = 0; \\ \rho_5 = 0; \\ \rho_6 = 0; \end{array} \right. \quad \left\{ \begin{array}{l} \rho_7 = 0; \\ \rho_8 = 0; \\ \rho_9 = 0,0061; \\ \rho_{10} = -0,0040; \\ \rho_{11} = 0,0007; \\ \rho_{12} = -0,0021; \end{array} \right. \quad \left\{ \begin{array}{l} \rho_{13} = 0,8782 + 0,4348i; \\ \rho_{14} = 0,8782 - 0,4348i; \\ \rho_{15} = 0,8937 + 0,4375i; \\ \rho_{16} = 0,8937 - 0,4375i; \\ \rho_{17} = 0,8899 + 0,4417i; \\ \rho_{18} = 0,8899 - 0,4417i; \end{array} \right. \quad \left\{ \begin{array}{l} \rho_{19} = 0,8675; \\ \rho_{20} = 1,0000; \\ \rho_{21} = 0,9552; \\ \rho_{22} = 0,9524; \\ \rho_{23} = 0,9524; \\ \rho_{24} = 0,9524. \end{array} \right. \quad (5-13)$$

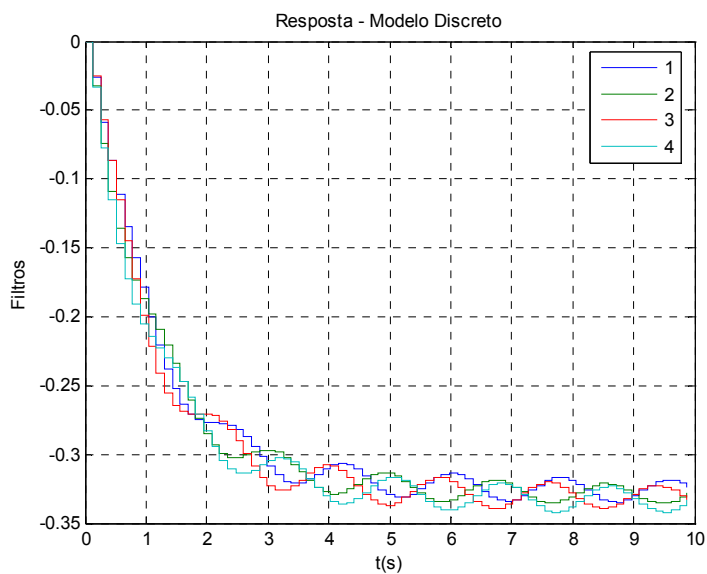


Figura 5-8: Malha com quatro nós: resposta do modelo de tempo discreto do caso 2.

A Figura 5-8 mostra degraus de largura maior devido ao período de amostragem utilizado. A precisão do modelo tende, portanto, a diminuir, pois o método de discretização envolve integração de funções obtidas via interpolação de pontos consecutivos. Este efeito pode ser observado pelo gráfico do erro entre 0 e 10s.

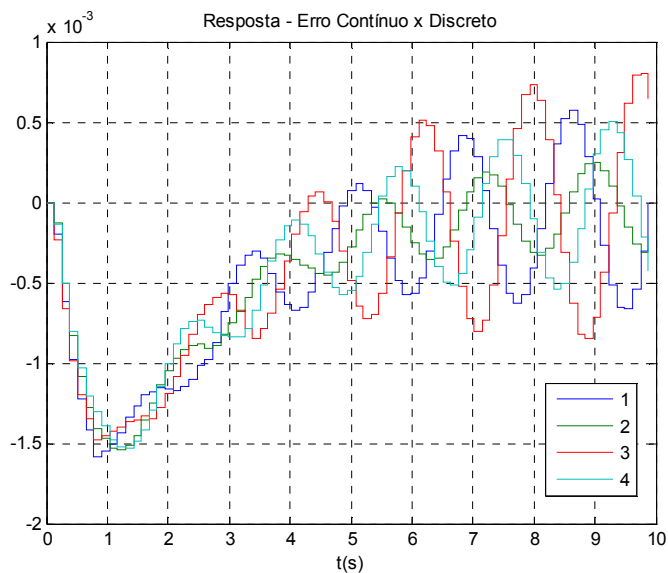


Figura 5-9: Malha com quatro nós: resposta do erro contínuo  $\times$  discreto do caso 2.

No entanto, o erro cometido entre o sistema discretizado e o de tempo contínuo tende a zero, visto que possuem a mesma frequência de sincronismo.

- *Sistema real*

A resposta do sistema real se mostra semelhante à do modelo de tempo discreto, de modo que se pode assumir que a estabilidade e o estado síncrono do modelo aproximado correspondem a malha real.

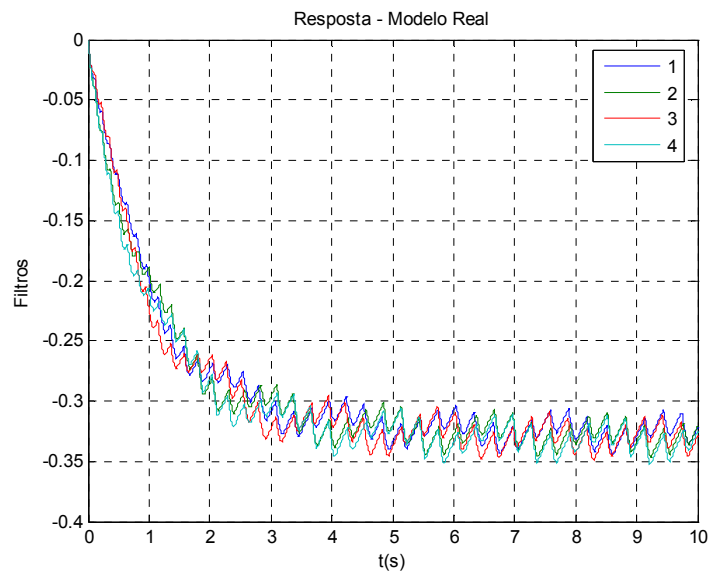


Figura 5-10: Malha com quatro nós: resposta do sistema real do caso 2.

A Figura 5-11 mostra que as saídas ponderadas do modelo discretizado tendem a acompanhar os valores médios das respostas reais. Porém, há uma maior discrepância em relação ao primeiro caso. Com o período de amostragem maior, a precisão do modelo aproximado tende a diminuir.



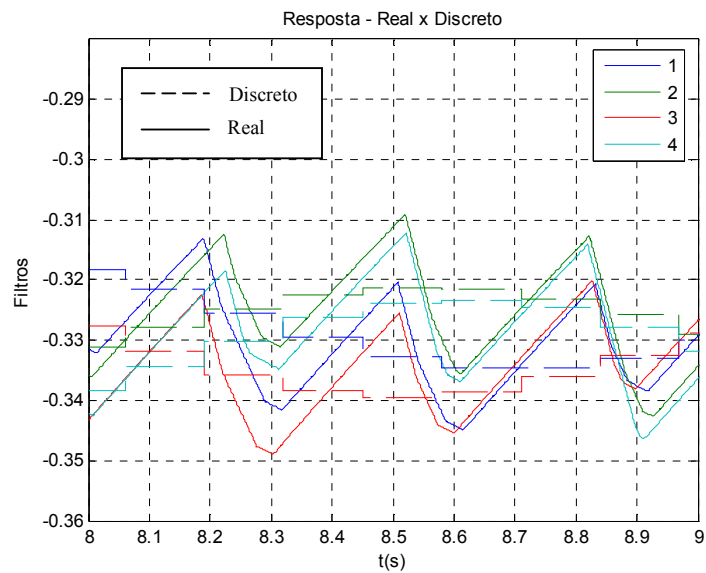


Figura 5-11: Malha com quatro nós: resposta real  $\times$  discreto do caso 2.

## 5.2 Redes com Cinco Nós

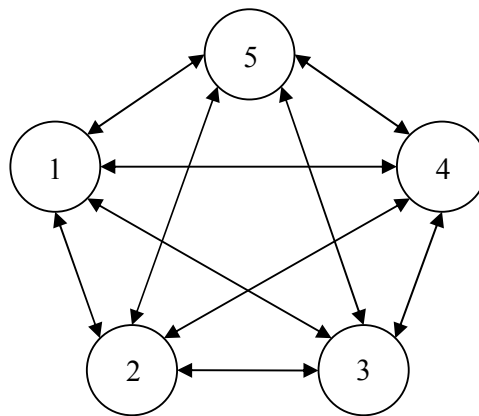


Figura 5-12: Rede com cinco nós mutuamente conectados.

A Figura 5-12 ilustra a rede com cinco nós. Esta rede possui dez conexões cujos atrasos iremos definir como:

$$\left\{ \begin{array}{l} \tau_{12} = \tau_1; \\ \tau_{13} = \tau_2; \\ \tau_{14} = \tau_3; \\ \tau_{15} = \tau_4; \\ \tau_{23} = \tau_5; \end{array} \right. \left\{ \begin{array}{l} \tau_{24} = \tau_6; \\ \tau_{25} = \tau_7; \\ \tau_{34} = \tau_8; \\ \tau_{35} = \tau_9; \\ \tau_{45} = \tau_{10}. \end{array} \right. \quad (5-14)$$

### 5.2.1 Caso 1

Tabela 5-3 – Resultados para rede de cinco nós – caso 1			
Frequência Central [Hz]	Ganho [Hz/V]	Filtro [V/rad]	Amostragem [s]
$f_1 = 8,8153$	$K_{o1} = 17,6307$	$\frac{s+2}{s^2+4s+3}$	$T_s = 0,74$
$f_2 = 8,9298$	$K_{o2} = 17,8596$		
$f_3 = 9,0443$	$K_{o3} = 18,0886$		
$f_4 = 9,1588$	$K_{o4} = 18,3176$		
$f_5 = 9,2733$	$K_{o5} = 18,5466$		
Atrasos [s]			
$\tau_1 = 0,0511$	$\tau_2 = 0,0117$	$\tau_3 = 0,0839$	$\tau_4 = 0,0500$
$\tau_5 = 0,0654$	$\tau_6 = 0,0813$	$\tau_7 = 0,0930$	$\tau_8 = 0,0764$
$\tau_9 = 0,0259$	$\tau_{10} = 0,0465$		
Respostas			
Frequência de sincronismo [Hz]		Autovalor Máximo Discreto	Saída dos Filtros em Regime [V]
Contínuo	Discreto		
6,6840	6,6840	1,0000	$u_{f1} = -0,1209$
			$u_{f2} = -0,1257$
			$u_{f3} = -0,1305$
			$u_{f4} = -0,1351$
			$u_{f5} = -0,1396$

- *Composição do estado*

A diferença que se percebe aqui em relação aos modelos de quatro nós é a presença de mais uma fase e os estados do quinto filtro, que nesse caso, é de segunda ordem.

$$x = [\theta_1 \quad v_{f1} \quad \dot{v}_{f1} \quad \theta_2 \quad v_{f2} \quad \dot{v}_{f2} \quad \theta_3 \quad v_{f3} \quad \dot{v}_{f3} \quad \theta_4 \quad v_{f4} \quad \dot{v}_{f4} \quad \theta_5 \quad v_{f5} \quad \dot{v}_{f5}]^T. \quad (5-15)$$

- *Sistema contínuo*

A novidade na construção do sistema de tempo contínuo fica por conta do maior número de atrasos que é preciso representar, aumentando a complexidade do problema.

$$\begin{aligned} \dot{x}(t) = & A_c x(t) + B_c u(t) + A_{c1} x(t - \tau_1) + A_{c2} x(t - \tau_2) + A_{c3} x(t - \tau_3) + \\ & A_{c4} x(t - \tau_4) + A_{c5} x(t - \tau_5) + A_{c6} x(t - \tau_6) + A_{c7} x(t - \tau_7) + \\ & A_{c8} x(t - \tau_8) + A_{c9} x(t - \tau_9) + A_{c10} x(t - \tau_{10}); \end{aligned} \quad (5-16)$$

$$y(t) = G_c x(t).$$



$$A_{c7} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c8} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad (5-17i)$$

$$A_{c9} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c10} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \end{bmatrix}; \quad (5-17j)$$

$${}^0A_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0,0199 & 0 & 0 \end{bmatrix}; \quad (5-17k)$$

$$G_c = \begin{bmatrix} 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 \end{bmatrix}. \quad (5-17l)$$

A Figura 5-13 fornece as saídas ponderadas do modelo de tempo contínuo. A simulação indica que o sistema converge para um estado estável, o que deverá ser comprovado pela discretização desse modelo.

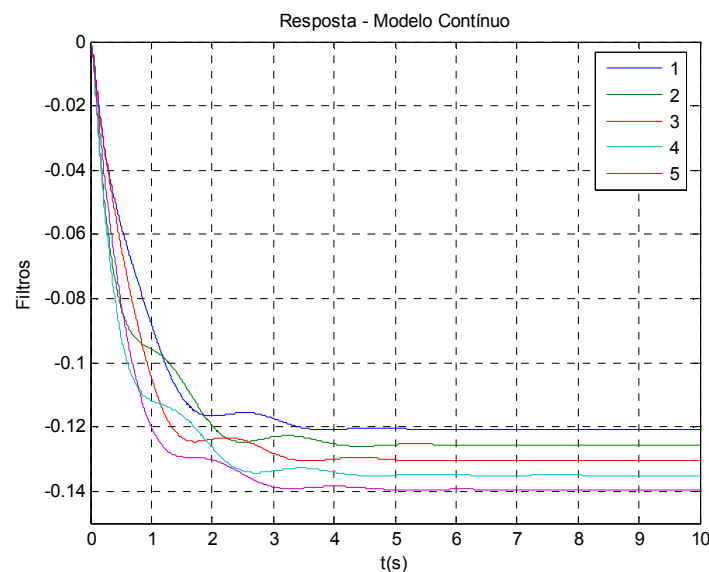


Figura 5-13: Malha com cinco nós: resposta do modelo de tempo contínuo do caso 1.

O filtro escolhido para os DPLLs, nesse caso, possui pólos mais à esquerda no plano complexo em comparação com os demais casos, o que torna as respostas dos nós mais rápidas. Para observar esse fato, ainda na Figura 5-13, basta notar que ao final da simulação, os valores das saídas ponderadas são praticamente constantes, havendo pequenas oscilações.

- *Sistema de tempo discreto*

O modelo discretizado possui uma forma semelhante à do segundo caso para redes de quatro nós. Porém as dimensões aqui tratadas são maiores como se observa em (5-18).

$$\begin{bmatrix} x_k \\ x_{k+1} \end{bmatrix} = \begin{bmatrix} 0_{15 \times 15} & I \\ {}^1A_d & 0_{A_d} \end{bmatrix} \begin{bmatrix} x_{k-1} \\ x_k \end{bmatrix} + \begin{bmatrix} 0_{15 \times 15} \\ {}^0B_d \end{bmatrix} u_k; \quad (5-18)$$

$$y_k = \begin{bmatrix} 0_{15 \times 15} & {}^0G_d \end{bmatrix}.$$

Seguem as matrizes da equação anterior detalhadas.

$${}^0A_d = \begin{bmatrix} -0,01 & 46,85 & 19,22 & 0,26 & 6,60 & 2,70 & 0,23 & 7,35 & 3,01 & 0,27 & 6,17 & 2,52 & 0,25 & 6,74 & 2,75 \\ -0,01 & 0,32 & 0,03 & 0,00 & 0,04 & 0,02 & 0,00 & 0,05 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 \\ 0,00 & -1,27 & -0,39 & 0,00 & 0,12 & 0,05 & -0,00 & 0,13 & 0,05 & 0,00 & 0,12 & 0,05 & -0,00 & 0,13 & 0,05 \\ 0,25 & 6,67 & 2,74 & -0,02 & 46,43 & 19,01 & 0,25 & 6,51 & 2,66 & 0,26 & 6,07 & 2,48 & 0,26 & 3,07 & 2,47 \\ 0,00 & 0,04 & 0,02 & -0,01 & 0,32 & 0,03 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 \\ -0,00 & 0,13 & 0,05 & 0,00 & -1,28 & -0,39 & -0,00 & 0,13 & 0,05 & -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 \\ 0,24 & 7,51 & 3,08 & 0,26 & 6,58 & 2,69 & -0,03 & 46,91 & 19,18 & 0,27 & 6,42 & 2,62 & 0,24 & 7,22 & 2,94 \\ 0,00 & 0,05 & 0,02 & 0,00 & 0,04 & 0,02 & -0,01 & 0,31 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 \\ -0,00 & 0,13 & 0,05 & -0,00 & 0,12 & 0,05 & 0,00 & -1,28 & -0,39 & -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 \\ 0,26 & 6,39 & 2,62 & 0,27 & 6,21 & 2,54 & 0,26 & 6,50 & 2,66 & -0,04 & 46,39 & 18,94 & 0,24 & 6,78 & 2,76 \\ 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & -0,01 & 0,31 & 0,02 & 0,00 & 0,04 & 0,02 \\ -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 & 0,00 & -1,28 & -0,39 & -0,00 & 0,12 & 0,05 \\ 0,26 & 7,05 & 2,89 & 0,28 & 6,28 & 2,57 & 0,24 & 7,39 & 3,02 & 0,26 & 6,85 & 2,80 & -0,04 & 46,67 & 19,02 \\ 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & 0,00 & 0,04 & 0,02 & -0,00 & 0,31 & 0,02 \\ -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 & -0,00 & 0,12 & 0,05 & 0,00 & -1,28 & -0,39 \end{bmatrix}; \quad (5-19a)$$

$${}^1A_d = \begin{bmatrix} 0_{15 \times 15} \end{bmatrix}; \quad (5-19b)$$

$${}^0B_d = \begin{bmatrix} 2,5936 & 0,3622 & 0,4000 & 0,3331 & 0,3609 \\ -0,0127 & 0,0023 & 0,0025 & 0,0021 & 0,0023 \\ -0,0328 & 0,0067 & 0,0070 & 0,0066 & 0,0067 \\ 0,3694 & 2,5480 & 0,3540 & 0,3274 & 0,3247 \\ 0,0023 & -0,0128 & 0,0022 & 0,0021 & 0,0021 \\ 0,0070 & -0,0329 & 0,0067 & 0,0065 & 0,0066 \\ 0,4159 & 0,3610 & 2,5522 & 0,3466 & 0,3867 \\ 0,0025 & 0,0022 & -0,0126 & 0,0022 & 0,0024 \\ 0,0070 & 0,0066 & -0,0325 & 0,0065 & 0,0066 \\ 0,3535 & 0,3406 & 0,3535 & 2,5035 & 0,3628 \\ 0,0022 & 0,0021 & 0,0022 & -0,0127 & 0,0022 \\ 0,0068 & 0,0066 & 0,0068 & -0,0325 & 0,0066 \\ 0,3904 & 0,3445 & 0,4020 & 0,3698 & 2,4982 \\ 0,0024 & 0,0021 & 0,0024 & 0,0022 & -0,0126 \\ 0,0069 & 0,0065 & 0,0068 & 0,0065 & -0,0322 \end{bmatrix}; \quad (5-19c)$$

$${}^0G_d = \begin{bmatrix} 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 \end{bmatrix}. \quad (5-19d)$$

O número de autovalores tende a ser maior aqui em comparação com as redes de quatro nós. Contudo, ainda vale o critério para os módulos dos autovalores. Como indica a Tabela 5-3 e (5-20), este é um sistema estável.

Autovalores  $s$  :

$$\left\{ \begin{array}{l} \rho_1 = 0; \\ \rho_2 = 0; \\ \rho_3 = 0; \\ \rho_4 = 0; \\ \rho_5 = 0; \\ \rho_6 = 0; \end{array} \right\} \left\{ \begin{array}{l} \rho_7 = 0; \\ \rho_8 = 0; \\ \rho_9 = 0; \\ \rho_{10} = 0; \\ \rho_{11} = 1,0000; \\ \rho_{12} = 0,3415; \end{array} \right\} \left\{ \begin{array}{l} \rho_{13} = -0,0557; \\ \rho_{14} = -0,3286 + 0,4563 i; \\ \rho_{15} = -0,3286 - 0,4563 i; \\ \rho_{16} = -0,3281 + 0,4336 i; \\ \rho_{17} = -0,3281 - 0,4336 i; \\ \rho_{18} = -0,3118 + 0,4444 i; \end{array} \right\} \left\{ \begin{array}{l} \rho_{19} = -0,3118 - 0,4444 i; \\ \rho_{20} = -0,3345 + 0,4600 i; \\ \rho_{21} = -0,3345 - 0,4600 i; \\ \rho_{22} = 0,1992; \\ \rho_{23} = 0,2012; \\ \rho_{24} = 0,2010; \end{array} \right\} \left\{ \begin{array}{l} \rho_{25} = 0,1999; \\ \rho_{26} = -0,0029; \\ \rho_{27} = 0,0001; \\ \rho_{28} = 0,0000; \\ \rho_{29} = -0,0003; \\ \rho_{30} = -0,0004. \end{array} \right. \quad (5-20)$$

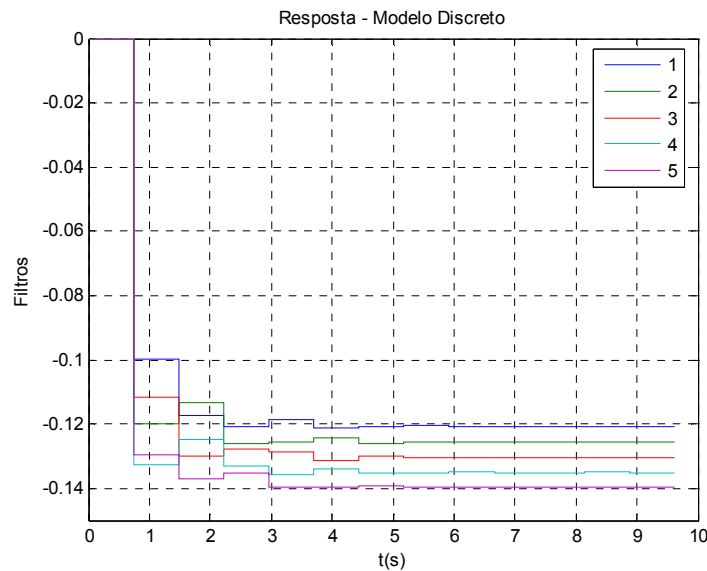


Figura 5-14: Malha com cinco nós: resposta do modelo de tempo discreto do caso 1.

Um valor alto foi propositadamente escolhido para o período de amostragem – como visto na Figura 5-14 – numa tentativa de se obter uma resposta menos precisa. O gráfico do erro a seguir mostra que, no início da simulação, o erro cometido pelo sistema discretizado teve o maior pico alcançado até o momento. Mesmo assim, observa-se que os valores de erro tendem a zero, como já previsto de acordo com as frequências de sincronismo idênticas – mostradas na Tabela 5-3 – para ambos os modelos, tanto de tempo contínuo quanto de tempo discreto.

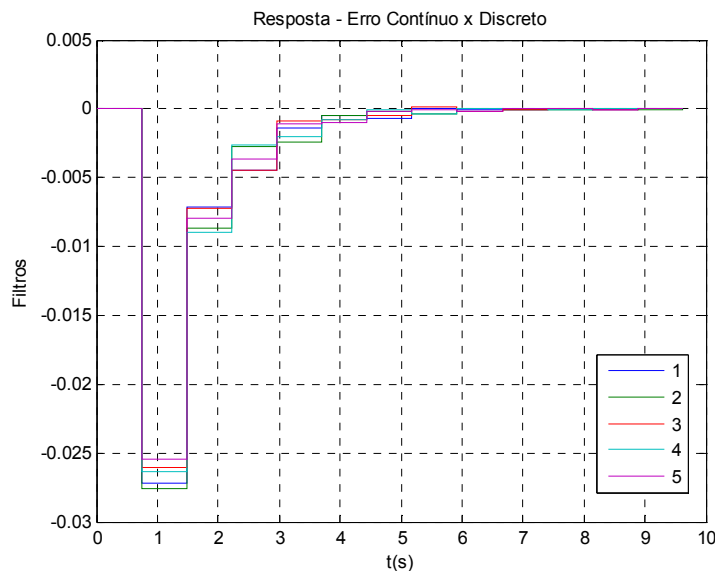


Figura 5-15: Malha com cinco nós: resposta do erro contínuo  $\times$  discreto do caso 1.

- *Sistema real*

As respostas do sistema real são, como esperado, semelhantes às respostas dos modelos, apesar das formas de onda obtidas estarem acrescidas de oscilações espúrias, características dos dispositivos reais. Como já mencionado, o filtro escolhido para esse caso possui pólos mais à esquerda no plano complexo, tornando mais rápidas as respostas dos nós. Isso faz com que os valores das saídas dos filtros apresentem oscilações de amplitude maior, que se pode ver, por exemplo, na Figura 5-16.

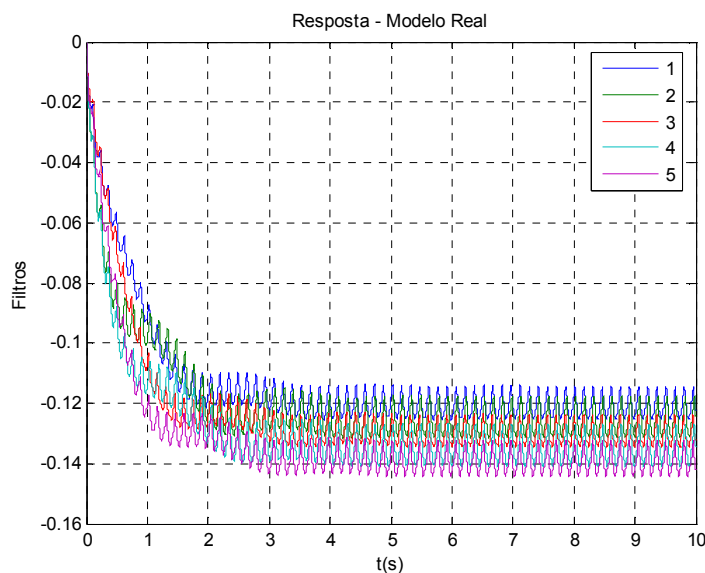


Figura 5-16: Malha com cinco nós: resposta do sistema real do caso 1.

O valor desses pólos faz com que o sistema se aproxime do estado síncrono mais rapidamente. Portanto, na Figura 5-17, a resposta do modelo de tempo discreto aproximado confunde-se com o valor médio das oscilações nas saída ponderadas reais.

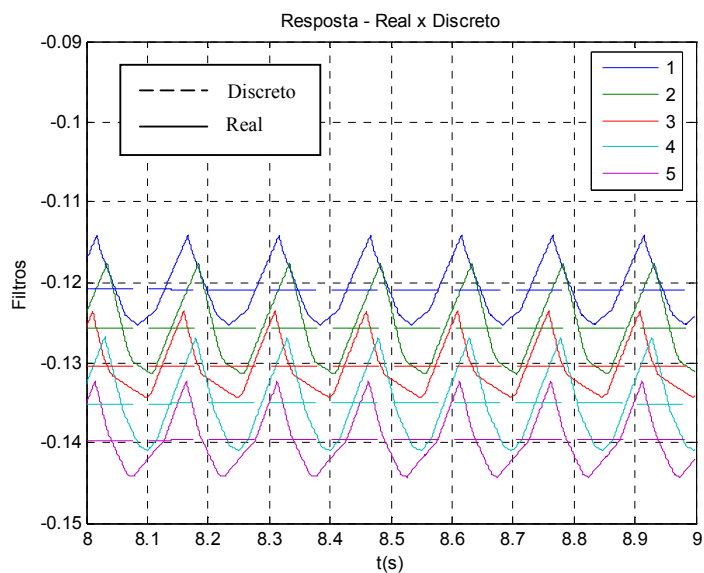


Figura 5-17: Malha com cinco nós: resposta do modelo real  $\times$  discreto do caso 1.



## 5.2.2 Caso 2

Tabela 5-4 – Resultados para rede de cinco nós – caso 2			
Frequência Central [Hz]	Ganho [Hz/V]	Filtro [V/rad]	Amostragem [s]
$f_1 = 10,5784$	$K_{o1} = 21,1568$	$\frac{1}{s^2 + 3s + 2}$	$T_s = 0,01$
$f_2 = 10,7158$	$K_{o2} = 21,4316$		
$f_3 = 10,8532$	$K_{o3} = 21,7063$		
$f_4 = 10,9906$	$K_{o4} = 21,9811$		
$f_5 = 11,1279$	$K_{o5} = 22,2559$		
Atrasos [s]			
$\tau_1 = 0,0955$	$\tau_2 = 0,0308$	$\tau_3 = 0,0646$	$\tau_4 = 0,0537$
$\tau_5 = 0,0902$	$\tau_6 = 0,0786$	$\tau_7 = 0,0511$	$\tau_8 = 0,0117$
$\tau_9 = 0,0839$	$\tau_{10} = 0,0500$		
Respostas			
Frequência de sincronismo [Hz]		Autovalor Máximo Discreto	Saída dos Filtros em Regime [V]
Contínuo	Discreto		
8,1515	-----	1,0029	$u_{f1} = \text{-----}$
			$u_{f2} = \text{-----}$
			$u_{f3} = \text{-----}$
			$u_{f4} = \text{-----}$
			$u_{f5} = \text{-----}$

- *Composição do estado*

A dimensão do estado no segundo caso para redes de cinco nós é igual a do primeiro. No modelo de tempo contínuo adotado, tal dimensão depende exclusivamente da ordem e da quantidade de nós na formação da rede.

$$x = [\theta_1 \quad v_{f1} \quad \dot{v}_{f1} \quad \theta_2 \quad v_{f2} \quad \dot{v}_{f2} \quad \theta_3 \quad v_{f3} \quad \dot{v}_{f3} \quad \theta_4 \quad v_{f4} \quad \dot{v}_{f4} \quad \theta_5 \quad v_{f5} \quad \dot{v}_{f5}]^T. \quad (5-21)$$

- *Sistema de tempo contínuo*

Nota-se aqui que a forma do sistema contínuo é idêntica ao do primeiro caso.

$$\begin{aligned} \dot{x}(t) = & A_c x(t) + B_c u(t) + A_{c1} x(t - \tau_1) + A_{c2} x(t - \tau_2) + A_{c3} x(t - \tau_3) + \\ & A_{c4} x(t - \tau_4) + A_{c5} x(t - \tau_5) + A_{c6} x(t - \tau_6) + A_{c7} x(t - \tau_7) + \\ & A_{c8} x(t - \tau_8) + A_{c9} x(t - \tau_9) + A_{c10} x(t - \tau_{10}); \end{aligned} \quad (5-22)$$

$$y(t) = G_c x(t).$$

Segue o detalhamento de (5-22):

$$A_c = \begin{bmatrix} {}^1A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & {}^2A_c & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^3A_c & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^4A_c & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^5A_c \end{bmatrix}; \quad (5-23a)$$



$$A_{c9} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} & 0_{3 \times 3} \end{bmatrix}; \quad A_{c10} = \begin{bmatrix} 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c \\ 0_{3 \times 3} & 0_{3 \times 3} & 0_{3 \times 3} & {}^0A_c & 0_{3 \times 3} \end{bmatrix}; \quad (5-23j)$$

$${}^0A_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0,0199 & 0 & 0 \end{bmatrix}; \quad (5-23k)$$

$$G_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \quad (5-23l)$$

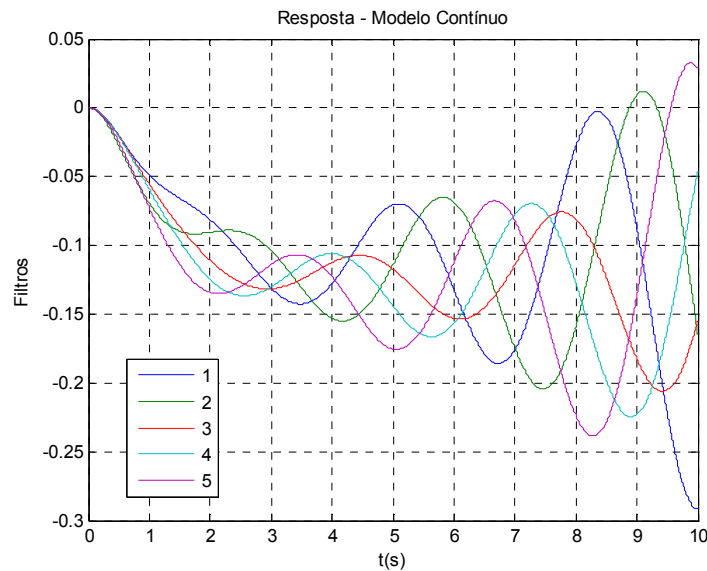


Figura 5-18: Malha com cinco nós: resposta do modelo de tempo contínuo do caso 2

A Figura 5-18 apresenta um comportamento inédito até agora. O sistema aparentemente tende à instabilidade, corroborada pelas oscilações crescentes dos valores das saídas ponderadas ao longo do tempo.

- *Sistema de tempo discreto*

Observando a Tabela 5-4, o maior módulo dos autovalores é maior que 1. Isso indica, de fato, que a rede não é estável próximo do estado síncrono. O pequeno valor do período de amostragem em relação aos atrasos da rede fez com que o sistema em (5-24) adquirisse grandes dimensões.

$$\begin{bmatrix} x_{k-9} \\ x_{k-8} \\ x_{k-7} \\ x_{k-6} \\ x_{k-5} \\ x_{k-4} \\ x_{k-3} \\ x_{k-2} \\ x_k \\ x_{k+1} \end{bmatrix} = \begin{bmatrix} 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} & 0_{1545} \\ 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & 0_{1545} & I & 0_{1545} & 0_{1545} \\ {}^{10}A_t & {}^9A_t & {}^8A_t & {}^7A_t & {}^6A_t & {}^5A_t & {}^4A_t & {}^3A_t & {}^2A_t & {}^1A_t & {}^0A_t & 0_{1545} \end{bmatrix} \begin{bmatrix} x_{k-10} \\ x_{k-9} \\ x_{k-8} \\ x_{k-7} \\ x_{k-6} \\ x_{k-5} \\ x_{k-4} \\ x_{k-3} \\ x_{k-2} \\ x_{k-1} \\ x_k \end{bmatrix} + \begin{bmatrix} 0_{1545} \\ {}^0B_t \end{bmatrix} u_k \quad (5-24)$$

$$y_k = \begin{bmatrix} 0_{15450} & {}^0C_t \end{bmatrix} \cdot$$

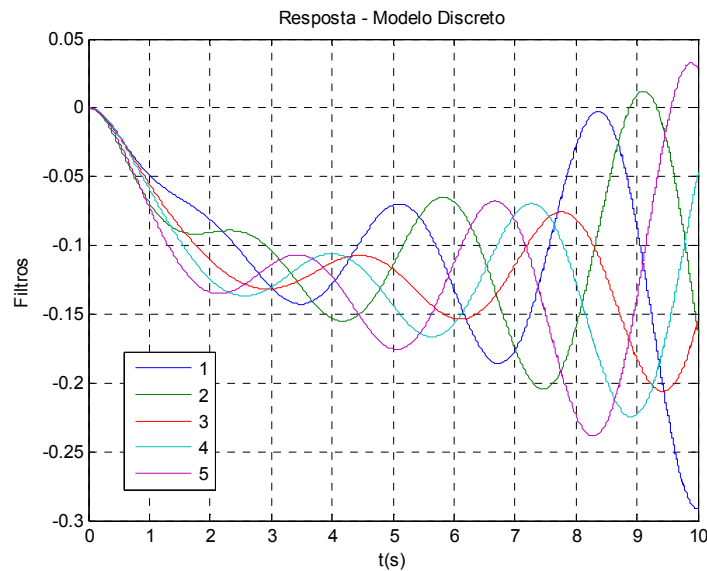


Figura 5-19: Malha com cinco nós: resposta do modelo de tempo discreto do caso 2

O período de amostragem utilizado foi escolhido pequeno, inclusive em relação aos pólos do filtro, de modo a induzir um comportamento mais preciso, com o menor erro possível. Porém, como mostrado pela Figura 5-20, o erro, apesar de mínimo no início da simulação –  $t = 0s$  – tende a crescer com o tempo.

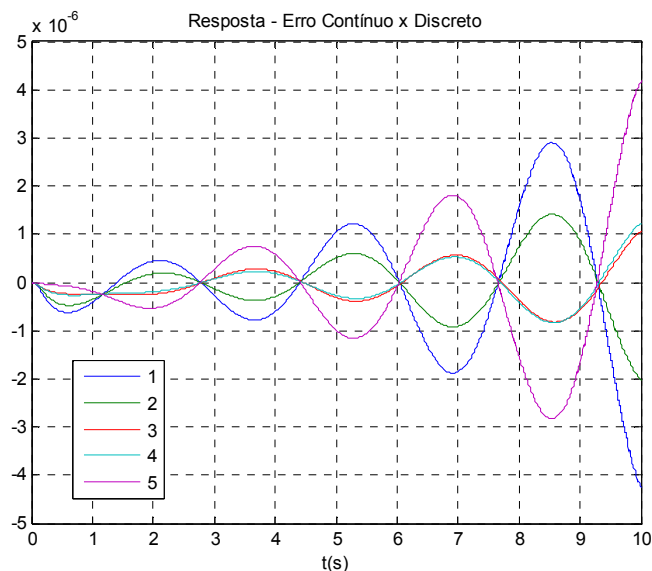


Figura 5-20: Malha com cinco nós: resposta do erro contínuo  $\times$  discreto do caso 2.

- *Sistema real*

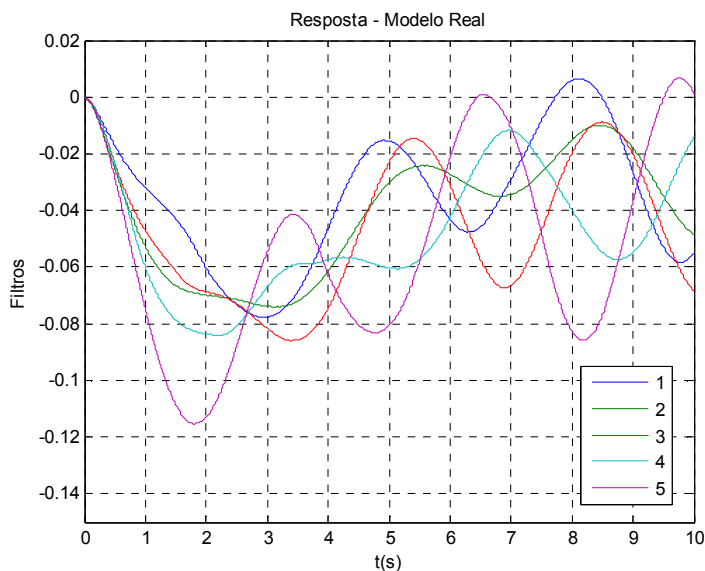


Figura 5-21: Malha com cinco nós: resposta do sistema real do caso 2.

A Figura 5-21 mostra as respostas ponderadas do sistema real. As oscilações indicam que não há tendência a um estado síncrono definido. Na Figura 5-22, os gráficos sobrepostos das respostas do modelo de tempo discreto aproximado e do sistema real já mostram discrepâncias logo no início das simulações. Isso fica evidente se se leva em conta que os

dispositivos reais possuem faixas de operação limitadas, sujeitas a não-linearidades. Portanto, as amplitudes das variações na resposta do sistema real ficam confinadas, enquanto que no modelo discretizado, as amplitudes crescem indefinidamente com o passar do tempo.

Apesar das diferenças nas simulações, deve-se atentar para o fato de que com os autovalores foi possível avaliar corretamente a estabilidade do sistema real. Quanto ao estado síncrono, o método apresentado no Capítulo 4 não pôde ser aplicado, já que não há como afirmar que exista uma frequência constante em regime, isto é, a frequência de sincronismo.

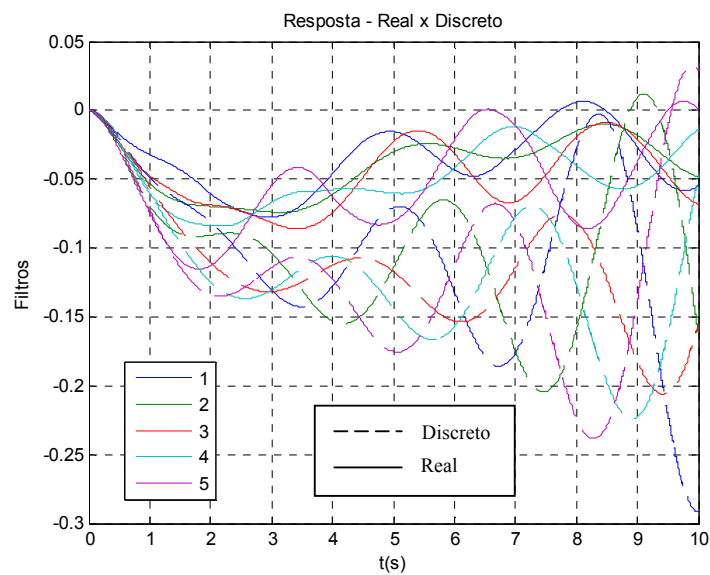


Figura 5-22: Malha com cinco nós: resposta real  $\times$  discreto do caso 2.

## Capítulo 6

### Conclusão

Este trabalho tem como principal objetivo apresentar uma nova forma de se analisar as redes de sincronismo mutuamente conectadas com atrasos. Para tanto, um método de discretização foi apresentado e que leva em consideração intervalos de tempo não nulos de propagação de sinais. A motivação para sua utilização foi suprimir o tratamento desses intervalos no modelo de tempo contínuo das redes de sincronismo. Os modelos de tempo discreto são baseados em atrasos, o que vem a facilitar o estudo do problema.

Trata-se, portanto, de representar um sistema real por um modelo aproximado que, nesse caso, é de tempo discreto. Como todo problema de modelagem, deseja-se obter um modelo que represente o sistema real o mais fielmente possível. Aqui, o primeiro passo foi obter um modelo de tempo contínuo para rede, o qual é discretizado. Esse processo de modelagem considera aproximações lineares do comportamento dos dispositivos que integram os DPLLs. Tais aproximações foram utilizadas nesse e, de forma semelhante, em outros trabalhos ([1]).

Assim, as contribuições desse trabalho podem ser divididas em duas vertentes, que são a precisão do modelo discretizado e a aplicação desse modelo às redes de sincronismo.

Quanto à precisão do modelo discretizado, tem-se o seguinte:

- o método adotado, por ser baseado em interpolações, faz com que o estado do modelo discretizado não coincida com o estado do modelo de tempo contínuo nos instantes múltiplos do período de amostragem  $T_s$ , sendo os atrasos necessariamente considerados; se não há atrasos, a discretização gera modelos cujos estados coincidem perfeitamente com os dos modelos de tempo contínuo;
- quanto menor o período de amostragem, maior é a precisão obtida, pois os pontos a serem interpolados estão mais próximos entre si, ressaltando que a não existência de

atrasos gera modelos de tempo discreto perfeitamente precisos com relação aos equivalentes de tempo contínuo;

- em comparação aos valores dos atrasos, se o período de amostragem for pequeno, a dimensão do modelo discretizado é grande, pois se leva em conta o número de períodos de amostragem que cabem dentro de um intervalo de tempo morto;
- há vantagem analítica na representação dos atrasos via modelos de tempo discreto.

Quanto à representação de tempo discreto das redes, tem-se o seguinte:

- os critérios adotados para analisar a estabilidade dos modelos discretizados apontam corretamente para a estabilidade do sistema real, para períodos de amostragem diversos;
- das aproximações utilizadas, não fazem parte aquelas que consideram as redes próximas ao estado síncrono, de modo que a análise da estabilidade utilizando modelos discretizados oferece vantagem em relação à análise via modelos de tempo contínuo;
- para sistemas estáveis, os modelos discretizados fornecem com precisão o estado síncrono da rede, independentemente do período de amostragem adotado;
- os modelos discretizados não podem substituir o estudo de sistemas reais no que se refere a simulações, visto que, em sistemas reais, as características físicas dos componentes notoriamente influenciam o comportamento das redes.



## Referências Bibliográficas

- [1] Piqueira, J.R.C.; Oliveira, M.Q.; Monteiro, L.H.A. – “Linear Approach for Synchronous State Stability in Fully Connected PLL” – Hindawi Publishing Corporation – Mathematical Problemas in Engineering – Volume 2008, Article ID 364084.
- [2] Carareto, R.; Orsatti, F.M.; Piqueira, J.R.C. – “Reachability of the Synchronous State in a Mutually-Connected PLL Network” – IEEE Transactions on Circuits and Systems I – Submitted.
- [3] Piqueira, J.R.C.; Oliveira, M.Q.; Monteiro, L.H.A. – “Synchronous State in a Fully Connected Phase-Locked Loop Network” – Hindawi Publishing Corporation – Mathematical Problemas in Engineering – Volume 2006, Article ID 52356.
- [4] Carareto, R.; Orsatti, F.M.; Piqueira, J.R.C. – “Multiple Synchronous States in Static Delay-free Mutually-Connected PLL Networks” – AEUE International Journal of Electronics and Communication – Submitted.
- [5] Piqueira, J.R.C. – “Uma Contribuição ao Estudo das Redes com Malhas de Sincronismo de Fase” – Tese apresentada para o concurso de Livre-Docência na área de Controle e Automação – São Paulo – 1997.
- [6] Orsatti, F.M. – “Redes Mutuamente Conectadas de DPLLs: Modelagem, Simulação e Otimização” – Tese de Doutorado – 2007.
- [7] Unzueta, M.V.R.; Piqueira, J.R.C. – “Discretization of a Continuous Time Delay System via Interpolation” – 40th Southeastern Symposium on System Theory – MB1.5 – 2008.
- [8] Jugo, J. – “Discretization of Continuous Time Delay Systems” – 15th IFAC World Congress on Automatic Control, IFAC, Barcelona, Spain, 2002.
- [9] Gupta, A.K.; Sinencio, E.S. – “State Space Approach to Design of Continuous Time Sigma Delta ADC with Delay in Feedback Path” – 49th IEEE International Midwest Symposium – Vol. 2 – pg. 664 – August, 2006.
- [10] del-Muro-Cuellar B.; Velasco-Villa M.; Puebla H.; Alvarez-Ramirez J. – “Control for Recycle Systems Based on a Discrete Time Model” Approximation – 16<sup>th</sup> IFAC World Congress in Prague.

- [11] Lindsey, W.C.; Chie, C.M. – “A Survey of Digital Phase-Locked Loop” – Proceedings of the IEEE – Vol. 69, No 4 – pg. 410 – April, 1981.
- [12] Teplinsky, A.; Feely O. – “Phase-Jitter Dynamics of Digital Phase-Locked Loops: Part II” – IEEE Transactions on Circuits and Systems - I: Fundamental Theory and Applications – Vol. 47, No. 4 – April, 2000 – pg. 458.
- [13] Osbourne, H.C. – “Stability Analysis of a Nth Power Digital Phase-Locked Loop - Part I: First Order DPLL” – IEEE Transactions on Communications – Vol. COM-28, No.8 – pg. 1343 – August, 1980.
- [14] Osbourne, H.C. – “Stability Analysis of a Nth Power Digital Phase-Locked Loop - Part II: Second- and Third-Order DPLL’s” – IEEE Transactions on Communications – Vol. COM-28 – pg. 1355 – August, 1980.
- [15] Oppenheim, A.V.; Schafer, R.W. – “Discrete Time Signal Processing” – 2<sup>nd</sup> Edition – Prentice Hall – 1999.
- [16] Ogata, K. – “Discrete-Time Control Systems” – Prentice Hall – 1986.
- [17] Ogata, K. – “Engenharia de Controle Moderno” – Prentice Hall – 4<sup>o</sup> edição.
- [18] Best, R. E. – “Phase-Locked Loops” – Ed. McGraw-Hill – 1984.
- [19] Åstrom, K.J. – “Introduction to Stochastic Control Theory” – Dover Publications – 1970.
- [20] Stengel, R.F – “Optimal Control and Estimation” – Dover Publications – 1994.
- [21] Kirk, D.E. – “Optimal Control Theory: An Introduction” – Dover Publications – 2004.
- [22] Kailath, T. – “Linear Systems” – Prentice Hall – 1979.
- [23] Ross, S.M. – “Introduction to Probability Models” – 8<sup>th</sup> Edition – Academic Press – 2002.
- [24] Bernstein D.S. – “Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear System Theory” – Princeton University Press – 2005.
- [25] Skogestad, S.; Postlethwaite, I. – “Multivariable Feedback Control: Analysis and Design” – Wiley-Interscience – 2<sup>nd</sup> Edition – 2005.
- [26] CD4046B Datasheet – Texas Instruments – Revised July 2003.

## Apêndice A

### Fórmulas de Integração

Abaixo encontram-se as fórmulas utilizadas nos algoritmos para cálculo das integrais das matrizes vistas em (4-12) e (4-23).

Antes de iniciar, mostramos a igualdade que pode ser encontrada em ([16]):

$$e^{At} = I + At + \frac{A^2}{2!}t^2 + \frac{A^3}{3!}t^3 + \frac{A^4}{4!}t^4 + \dots ;$$

$$e^{At} = \sum_{i=0}^{\infty} \frac{A^i}{i!} t^i. \quad (\text{A-1})$$

A partir da igualdade acima, obtemos os próximos resultados para os dois tipos diferentes de integral vistos anteriormente.

#### A.1 Integração 1

A integral que desejamos estimar aqui é da forma:

$$\int_c^d e^{-A\mu} d\mu. \quad (\text{A-2})$$

De (A-1), temos:

$$\begin{aligned} \int_c^d \sum_{i=0}^{\infty} \frac{A^i}{i!} \mu^i d\mu &= \int_c^d \left( I + A\mu + \frac{A^2}{2!} \mu^2 + \frac{A^3}{3!} \mu^3 + \dots \right) d\mu = \\ &= \left( I\mu + \frac{A}{2} \mu^2 + \frac{A^2}{3 \cdot 2!} \mu^3 + \frac{A^3}{4 \cdot 3!} \mu^4 + \dots \right)_c^d = \\ &= \sum_{i=0}^{\infty} \frac{A^i}{(i+1)!} (d^{(i+1)} - c^{(i+1)}). \end{aligned} \quad (\text{A-3})$$

Já para integrais da forma

$$\int_c^d e^{-A\mu} d\mu. \quad (\text{A-4})$$

devemos proceder de forma um pouco diferente. De (A-1), temos:

$$e^{-At} = I - At + \frac{A^2}{2!}t^2 - \frac{A^3}{3!}t^3 + \frac{A^4}{4!}t^4 - \frac{A^5}{4!}t^5 + \dots ;$$

$$e^{-At} = \sum_{i=0}^{\infty} \frac{A^i}{i!} (-t)^i. \quad (\text{A-5})$$

Substituindo (A-5) em (A-4):

$$\int_c^d \sum_{i=0}^{\infty} \frac{A^i}{i!} (-\mu)^i d\mu = \int_c^d \left( I - A\mu + \frac{A^2}{2!} \mu^2 - \frac{A^3}{3!} \mu^3 + \dots \right) d\mu =$$

$$= \left( I\mu - \frac{A}{2} \mu^2 + \frac{A^2}{3 \cdot 2!} \mu^3 - \frac{A^3}{4 \cdot 3!} \mu^4 + \dots \right)_c^d =$$

$$= \sum_{i=0}^{\infty} (-1)^{i+1} \frac{A^i}{(i+1)!} (d^{(i+1)} - c^{(i+1)}). \quad (\text{A-6})$$

## A.2 Integração 2

A integral que desejamos estimar aqui é um pouco mais complicada e tem a forma:

$$\int_c^d e^{-A\mu} \mu d\mu. \quad (\text{A-7})$$

Substituindo (A-5) em (A-7):

$$\int_c^d \sum_{i=0}^{\infty} \frac{A^i}{i!} (-\mu)^i \mu d\mu = \int_c^d \left( I - A\mu + \frac{A^2}{2!} \mu^2 - \frac{A^3}{3!} \mu^3 + \dots \right) \mu d\mu =$$

$$= \int_c^d \left( I\mu - A\mu^2 + \frac{A^2}{2!} \mu^3 - \frac{A^3}{3!} \mu^4 + \dots \right) d\mu =$$

$$= \left( \frac{I}{2} \mu^2 - \frac{A}{3} \mu^3 + \frac{A^2}{4 \cdot 2!} \mu^4 - \frac{A^3}{5 \cdot 3!} \mu^5 + \dots \right)_c^d =$$

$$= \left( 1 \frac{I}{2!} \mu^2 - 2 \frac{A}{3!} \mu^3 + 3 \frac{A^2}{4!} \mu^4 - 4 \frac{A^3}{5!} \mu^5 + \dots \right)_c^d =$$

$$= \sum_{i=0}^{\infty} (i+1) \frac{A^i}{(i+2)!} \left( (-d)^{(i+2)} - (-c)^{(i+2)} \right). \quad (\text{A-8})$$

Quando da implementação dos algoritmos, é preciso estabelecer um ponto de parada das iterações, pois não é possível se obter o valor exato das somatórias com infinito número de termos. Este ponto de parada pode ser uma análise da variação dos elementos da somatória a cada iteração.

## **Apêndice B**

### **Listagens**

Abaixo encontram-se as listagens dos programas em MATLAB utilizados para gerar as redes contínuas e discretizá-las. Todas elas foram depuradas e utilizadas na versão 7, release 14.

## B.1 Função SIMREDE

```

%*****
% Função SIMREDE
% Cria, discretiza e simula rede de PLL's
%
% Entrada
% u      = Entrada do sistema
% Tsim   = Tempo de simulação
% Ts     = Tempo de amostragem
% Ko     = Vetor linha com os ganhos dos DPLL's
% Td     = Vetor linha com os atrasos entre os nós
% NumF   = Matriz cujas linhas são os numeradores dos filtros de cada nó
% DenF   = Matriz cujas linhas são os denominadores dos filtros de cada nó
%
% Saída
% Kd     = Modelo discreto no espaço de estados
% xd0    = Condição inicial discreta
% Ac, Bc, Cc e Gc são matrizes do sistema contínuo
% dx/dt = Ac*x + Bc*u + Cc*x(t - Td)
% y      = Gc*x
% xc0    = Condição inicial do sistema contínuo
% yreg   = Saída do sistema em regime
% xreg   = Estado do sistema em regime
%*****

function [Kd, xd0, Ac, Bc, Cc, Gc, xc0, yreg, xreg] = simrede(u, Tsim, Ts, Ko, Td, NumF, DenF,
tipo)

%Inicia contagem do tempo de execução dos algoritmos
tic;

%Cria matrizes do modelo contínuo da rede
[Ac, Bc, Cc, Gc, dimpll] = criarede(u, Ko, Td, NumF, DenF, tipo);

%Cria condições iniciais nulas para rede
xc0 = zeros(size(Ac,2),1);

%Discretiza rede
[Kd, xd0] = discretiza(Ac, Bc, Cc, 0, Gc, 0, Td, Ts, xc0);

%Determina o estado de regime e calcula autovalores
[yreg, xreg, autoval] = regime(Kd, u, Ts, dimpll);

%Encerra e mostra o tempo de execução dos algoritmos
toc

%Simulação
Tpassos = floor(Tsim/Ts); %Arredonda para o maior número inteiro maior que a divisão
x = xd0; %Valor inicial de x
for k=1:Tpassos+1
    t(k) = Ts*(k-1); %Vetor tempo
    y(:,k) = Kd.c*x; %Saída do sistema
    x = Kd.a*x + Kd.b*u; %Sistema discreto
end

if 1
    %figure(1);
    hold;
    cor = 7;
    for k=1:size(y,1)
        if cor == 7
            color = 'r-';
            cor = 1;
        elseif cor == 1
            color = 'g-';
            cor = 2;
        elseif cor == 2
            color = 'b-';
            cor = 3;
        end
    end
end

```

```
elseif cor == 3
    color = 'c-';
    cor = 4;
elseif cor == 4
    color = 'm-';
    cor = 5;
elseif cor == 5
    color = 'y-';
    cor = 6;
elseif cor == 6
    color = 'k-';
    cor = 7;
end

stairs(t,y(k,:),color);
end
grid;
ylabel('y');
xlabel('t');
title('Simulação');
end

return;
```



## B.2 Função CRIAREDE

```

%*****
% Função CRIAREDE
% Cria rede de PLL's
%
% Entrada
% u      = Entrada do sistema
% Ko     = Vetor linha com os ganhos dos DPLL's
% Td     = Vetor linha com os atrasos entre os nós
% NumF   = Matriz cujas linhas são os numeradores dos filtros de cada nó
% DenF   = Matriz cujas linhas são os denominadores dos filtros de cada nó
% tipo   = Opção de filtros diferentes para cada detector ou não
%
% Saida
% Ac, Bc, Cc e Gc são matrizes do sistema contínuo
% dx/dt = Ac*x + Bc*u + Cc*x(t - Td)
% y = Gc*x
%*****

function [Ac, Bc, Cc, Gc, dimpll] = criarede(u, Ko, Td, NumF, DenF, tipo)

nnos = size(Ko, 2);      %Número de nós da rede
ndelays = (nnos - 1)*nnos/2; %Número de atrasos possíveis

if ndelays ~= size(Td, 2)
    'Quantidade de delays deve ser'
    ndelays
    return;
end

if tipo == 's' %Fase simples
    for k = 1:nnos
        tam = size(DenF(k, :), 2);
        dimf(k) = tam - 1;
        for m = 1:tam
            if DenF(k, m) ~= 0
                break;
            else
                dimf(k) = dimf(k) - 1;      %Tamanho de cada filtro dentro do pll
            end
        end

        dimpll(k) = dimf(k) + 1; %Tamanho total de cada pll

        if k == 1
            indpll(k) = 1;      %Índice do pll dentro da matriz
        else
            indpll(k) = 1 + sum(dimpll(1:(k - 1))); %Índice do pll dentro da matriz
        end
    end

    for k = 1:nnos
        Fr = ssfiltro(NumF(k,:), DenF(k,:)); %Gera filtro de realimentação

        Ac(indpll(k):indpll(k) + dimpll(k) - 1, indpll(k):indpll(k) + dimpll(k) - 1) = [0
        2*pi*Ko(k)*Fr.c; (-1/(4*pi))*Fr.b Fr.a];

        %Gera entrada com frequência central
        Bc(indpll(k):indpll(k) + dimpll(k) - 1, k) = [2*pi; zeros(dimf(k), 1)];

        %Gera entrada com frequência central e erro do detector de fases
        %Bc(indpll(k):indpll(k) + dimpll(k) - 1, (k - 1)*2 + 1:(k - 1)*2 + 2) = [[2*pi;
        zeros(dimf(k), 1)] [0; Fr.b]];

        %Gera saída com filtros
        Gc(k, indpll(k):indpll(k) + dimpll(k) - 1) = [0 Fr.c];

        %Gera saída com fases
        %Gc(k, indpll(k):indpll(k) + dimpll(k) - 1) = [1 zeros(1, dimf(k))];
    end
end

```

```

end

Cc = zeros(size(Ac,1), size(Ac,2), ndelays); %Gera realimentação

indel = 0; %Índice do delay
for k = 1:(nnos - 1)
    for l = (k + 1):nnos
        indel = indel + 1; %Incrementa índice de delays

        Fr = ssfiltro(NumF(k,:), DenF(k,:)); %Gera filtro de realimentação
        Fl = ssfiltro(NumF(l,:), DenF(l,:)); %Gera filtro de realimentação

        Cc(indp11(k) + 1:indp11(k) + dimf(k), indp11(l), indel) = (1/(4*pi*(nnos -
1)))*Fr.b;
        Cc(indp11(l) + 1:indp11(l) + dimf(l), indp11(k), indel) = (1/(4*pi*(nnos -
1)))*Fl.b;
    end
end
elseif tipo == 'c' %Fase composta
    for k = 1:nnos
        for l = 1:(nnos - 1)
            tam = size(DenF((k - 1)*(nnos - 1) + 1, :), 2);
            dimf(k,l) = tam - 1;
            for m = 1:tam
                if DenF((k - 1)*(nnos - 1) + 1, m) ~= 0
                    break;
                else
                    dimf(k,l) = dimf(k,l) - 1; %Tamanho de cada filtro dentro do pll
                end
            end
            end

            if l == 1
                indf(k,l) = 2; %Índice do filtro dentro da matriz
            else
                indf(k,l) = 2 + sum(dimf(k,1:(l - 1))); %Índice do filtro dentro da matriz
            end
        end

        dimpll(k) = 1 + sum(dimf(k,:)); %Tamanho total de cada pll

        if k == 1
            indp11(k) = 1; %Índice do pll dentro da matriz
        else
            indp11(k) = 1 + sum(dimpll(1:(k - 1))); %Índice do pll dentro da matriz
        end
    end

    for k = 1:nnos
        Ac(indp11(k), indp11(k)) = 0;

        %Gera entrada com frequência central
        Bc(indp11(k):indp11(k) + dimpll(k) - 1, k) = [2*pi; zeros(dimpll(k) - 1,1)];

        %Gera entrada com frequência central e erro do detector de fases
        %Bc(indp11(k):indp11(k) + dimpll(k) - 1, (k - 1)*nnos + 1) = [2*pi; zeros(dimpll(k) -
1,1)];
        for l = 1:(nnos - 1)
            Fl = ssfiltro(NumF((k - 1)*(nnos - 1) + 1,:), DenF((k - 1)*(nnos - 1) + 1,:));
        %Gera filtro de realimentação
            Ac(indp11(k), indp11(k) + indf(k,l) - 1:indp11(k) + indf(k,l) - 2 + dimf(k,l)) =
[2*pi*Ko(k)*Fl.c/(nnos - 1)];
            Ac(indp11(k) + indf(k,l) - 1:indp11(k) + indf(k,l) + dimf(k,l) - 2, indp11(k)) = -
(1/(4*pi))*Fl.b;
            Ac(indp11(k) + indf(k,l) - 1:indp11(k) + indf(k,l) + dimf(k,l) - 2, indp11(k) +
indf(k,l) - 1:indp11(k) + indf(k,l) - 2 + dimf(k,l)) = Fl.a;

            %Gera entrada com frequência central e erro do detector de fases
            %Bc(indp11(k) + indf(k,l) - 1:indp11(k) + indf(k,l) + dimf(k,l) - 2, (k - 1)*nnos
+ 1 + 1) = [Fl.b];

            %Gera saída com filtros separados
            Gc((k - 1)*(nnos - 1) + 1, indp11(k) + indf(k,l) - 1:indp11(k) + indf(k,l) - 2 +
dimf(k,l)) = [Fl.c];

            %Gera saída com filtros juntos (como em 'fss')

```

```

                %Gc(k, indp11(k) + indf(k,1) - 1:indp11(k) + indf(k,1) - 2 + dimf(k,1)) =
[(1/(nnos - 1))*Fl.c];
            end

            %Gera saída com fases
            %Gc(k, indp11(k):indp11(k) + dimp11(k) - 1) = [1 zeros(1, dimp11(k) - 1)];
        end

        Cc = zeros(size(Ac,1), size(Ac,2), ndelays);    %Gera realimentação

        indel = 0; %Índice de delays
        for k = 1:(nnos - 1)
            for l = (k + 1):nnos
                indel = indel + 1; %Incrementa índice de delays

                Fk = ssfiltro(NumF((k - 1)*(nnos - 1) + l - 1,:), DenF((k - 1)*(nnos - 1) + l -
1,:)); %Gera filtro de realimentação
                Cc(indp11(k) + indf(k, l - 1) - 1:indp11(k) + indf(k, l - 1) + dimf(k, l - 1) - 2,
indp11(l), indel) = (1/(4*pi))*Fk.b;

                Fl = ssfiltro(NumF((l - 1)*(nnos - 1) + k,:), DenF((l - 1)*(nnos - 1) + k,:));
                %Gera filtro de realimentação
                Cc(indp11(l) + indf(l, k) - 1:indp11(l) + indf(l, k) + dimf(l, k) - 2, indp11(k),
indel) = (1/(4*pi))*Fl.b;
            end
        end
    end

    %Calcula frequência de sincronismo
    a0 = NumF(1,size(NumF,2));
    b0 = DenF(1,size(DenF,2));

    somatd = sum(Td);
    somawk = 0;
    somak = 0;
    for k = 1:nnos
        somawk = somawk + 2*pi*u(k)/(Ko(k)/(4*pi));
        somak = somak + 1/(Ko(k)/(4*pi));
    end

    'Frequência de sincronismo - Modelo Contínuo'
    ws = somawk/(somak + ((4*pi*a0)/(b0*(nnos-1)))*somatd);
    fs = ws/(2*pi)

    %*****
    % Função para geração do filtro no espaço de estados *****
    function F = ssfiltro(NumF, DenF)

    [a, b, c, d] = tf2ss(NumF, DenF);

    n = size(a,1); %Dimensão do filtro
    for k = 1:n
        T(k,n-k+1) = 1;
    end

    a = T*a*T; %Mudança de base
    b = T*b;
    c = c*T;

    F = ss(a, b, c, d);
    return;

```

## B.3 Função DISCRETIZA

```

%*****
% Função DISCRETIZA
% Discretiza sistema contínuo com atraso
%
% Entrada
% Ac, Bc, Cc, Dc, Gc e Hc são matrizes do sistema contínuo
% dx/dt = Ac*x + Bc*u + Cc*x(t - Td) + Dc*u(t - Td)
% y = Gc*x + Hc*x(t - Td)
% Td = Vetor linha com os atrasos
% Ts = Período de amostragem
% xc0 = Condição inicial contínua
%
% Saída
% Kd = Modelo discreto no espaço de estados
% xd0 = Condição inicial discreta
% n = Vetor com os valores de ni
%*****

function [Kd, xd0, n] = discretiza(Ac, Bc, Cc, Dc, Gc, Hc, Td, Ts, xc0)

%Dimensão do sistema contínuo
dim = size(Ac,1);

%Define condição inicial contínua xc0 caso não esteja definida
if (exist('xc0')== 0)
    xc0 = zeros(dim,1);
else
    if (size(xc0,1)~=dim) || (size(xc0,2)~=1)
        xc0 = zeros(dim,1);
    end
end

%Atribui condição inicial contínua à condição inicial discreta
xd0 = xc0;

%Número de atrasos do sistema
numdly = size(Td,2);

An = expm(Ac*Ts);
Bn = integexp(Ac, Ts, 0)*Bc;
Gn = Gc;

%Encontra o valor de Phi onde Td = n*Ts + Phi
for a = 1:numdly
    n(1,a) = fix(Td(a)/Ts);
    Phi(1,a) = Td(a) - n(1,a)*Ts;
end
maxn = max(n); %Valor máximo de n

%Cria sistema para tratar estados atrasados
if (max(max(max(Cc ~= 0))) > 0) || (max(max(max(Hc ~= 0))) > 0)
    Atemp = zeros(dim); %Matriz de zeros com a mesma dimensão de Ac
    Am = [zeros(dim*(maxn + 1), (dim*(maxn + 1))) zeros(dim*(maxn + 1), dim);
zeros(dim, (dim*(maxn + 1))) An];
    Bm = [zeros(dim*(maxn + 1), size(Bc,2)); Bn];
    Gm = [zeros(size(Gc,1), dim*(maxn + 1)) Gn];
    xm0 = [zeros(dim*(maxn + 1), 1); xc0];

    %Altera sistema caso haja realimentação de estados atrasados
    if max(max(Cc ~= 0)) > 0
        for a = 1:numdly
            %Gera contantes
            Cte0 = expm(Ac*Ts);
            Cte1 = integinvexplin(Ac, Ts, Phi(1,a));
            Cte2 = integinvexp(Ac, Ts, Phi(1,a));
            Cte3 = integinvexplin(Ac, Phi(1,a), 0);
            Cte4 = integinvexp(Ac, Phi(1,a), 0);
        end
    end
end

```

```

%Multiplicador de x(k-n+1)
Pd(:, :, a) = (1/Ts)*Cte0*(+Cte1 - Cte2*Phi(1,a))*Cc(:, :, a);
%Multiplicador de x(k-n)
Qd(:, :, a) = (1/Ts)*Cte0*(-Cte1 + Cte2*(Ts + Phi(1,a)) + Cte3 + Cte4*(Ts -
Phi(1,a)))*Cc(:, :, a);
%Multiplicador de x(k-n-1)
Rd(:, :, a) = (1/Ts)*Cte0*(-Cte3 + Cte4*Phi(1,a))*Cc(:, :, a);

%Cria sistema: Am e Bm (x(k+1) = Am*x(k) + Bm*u(k))
Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 0)+1):(dim*(maxn - n(a)
+ 1))) = Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 0)+1):(dim*(maxn - n(a) +
1))) + Rd(:, :, a);
Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 1)+1):(dim*(maxn - n(a)
+ 2))) = Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 1)+1):(dim*(maxn - n(a) +
2))) + Qd(:, :, a);
if n(1,a) >= 1
Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 2)+1):(dim*(maxn -
n(a) + 3))) = Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(maxn - n(a) + 2)+1):(dim*(maxn -
n(a) + 3))) + Pd(:, :, a);
else
Atemp = Atemp + Pd(:, :, a);
end
end

%Caso exista algum n = 0, multiplica as linhas das matrizes com o
%inverso do multiplicador de x(k+1), a saber, inv(I - Atemp)
if min(n) < 1
for a = 1:maxn+2
Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(a - 1)+1):dim*a) = inv(eye(dim) -
Atemp)*Am((dim*(maxn + 1)+1):(dim*(maxn + 2)), (dim*(a - 1)+1):dim*a);
end
Bm((dim*(maxn + 1)+1):(dim*(maxn + 2)), 1:size(Bc,2)) = inv(eye(dim) -
Atemp)*Bm((dim*(maxn + 1)+1):(dim*(maxn + 2)), 1:size(Bc,2));
end
end

%Altera sistema caso haja saídas atrasadas
if max(max(Hc ~= 0)) > 0
for a = 1:numdly
%Cálculo a partir de x(k-n-1)
Gm(1:size(Gc,1), (dim*(maxn - n(a) + 0)+1):(dim*(maxn - n(a) + 1))) =
Gm(1:size(Gc,1), (dim*(maxn - n(a) + 0)+1):(dim*(maxn - n(a) + 1))) +
Hc(:, :, a)*(eye(dim)*Phi(1,a)/Ts);
%Cálculo a partir de x(k-n)
Gm(1:size(Gc,1), (dim*(maxn - n(a) + 1)+1):(dim*(maxn - n(a) + 2))) =
Gm(1:size(Gc,1), (dim*(maxn - n(a) + 1)+1):(dim*(maxn - n(a) + 2))) + Hc(:, :, a)*(eye(dim) -
eye(dim)*Phi(1,a)/Ts);
end
end

%Preenche diagonal com matrizes-identidade
for a = 1:maxn+1
Am((dim*(a - 1)+1):dim*a, dim*a + 1:dim*(a + 1)) = eye(dim);
end
else
Am = An;
Bm = Bn;
Gm = Gn;
xm0 = xc0;
end

%Cria sistema para entradas atrasadas
if max(max(max(Dc ~= 0)) > 0
Ad = [zeros(size(Bm,2)*(maxn + 1)) zeros(size(Bm,2)*(maxn + 1), size(Am,2));
zeros(size(Bm,1), size(Bm,2)*(maxn + 1)) Am];
Bd = [zeros(size(Bm,2)*maxn, size(Bm,2)); eye(size(Bm,2)); Bm];
Gd = [zeros(size(Gm,1), size(Bm,2)*(maxn + 1)) Gm];
xd0 = [zeros(size(Bm,2)*(maxn + 1), 1); xm0];

for a = 1:numdly
%Cria matrizes do modelo discreto para entradas atrasadas (já
%contempla o caso em que há n = 0
if (max(max(max(Cc ~= 0)) > 0) || (max(max(max(Hc ~= 0)) > 0)
%Multiplicador de u(k-n)

```

```

        Vd((dim*(maxn + 1)+1):(dim*(maxn + 2)),1:size(Bm,2),a) = inv(eye(dim) -
Atemp)*integexp(Ac, Ts-Phi(1,a), 0)*Dc(:, :, a);
        %Multiplicador de u(k-n-1)
        Wd((dim*(maxn + 1)+1):(dim*(maxn + 2)),1:size(Bm,2),a) = inv(eye(dim) -
Atemp)*integexp(Ac, Ts, Ts-Phi(1,a))*Dc(:, :, a);
    else
        Cte0 = expm(Ac*Ts);
        %Multiplicador de u(k-n)
        Vd(:, :, a) = Cte0*integexp(Ac, Ts, Phi(1,a))*Dc(:, :, a);
        %Multiplicador de u(k-n-1)
        Wd(:, :, a) = Cte0*integexp(Ac, Phi(1,a), 0)*Dc(:, :, a);

    end

    Ad((size(Bm,2)*(maxn + 1)+1):(size(Bm,2)*(maxn + 1)+size(Am,1)), (size(Bm,2)*(maxn -
n(1,a))+1):(size(Bm,2)*(maxn - n(1,a) + 1))) = Ad((size(Bm,2)*(maxn + 1)+1):(size(Bm,2)*(maxn
+ 1)+size(Am,1)), (size(Bm,2)*(maxn - n(1,a))+1):(size(Bm,2)*(maxn - n(1,a) + 1))) + Wd(:, :, a);
    if n(1,a) >= 1
        Ad((size(Bm,2)*(maxn + 1)+1):(size(Bm,2)*(maxn + 1)+size(Am,1)), (size(Bm,2)*(maxn
+ 1 - n(1,a))+1):(size(Bm,2)*(maxn - n(1,a) + 2))) = Ad((size(Bm,2)*(maxn +
1)+1):(size(Bm,2)*(maxn + 1)+size(Am,1)), (size(Bm,2)*(maxn + 1 - n(1,a))+1):(size(Bm,2)*(maxn
- n(1,a) + 2))) + Vd(:, :, a);
    else
        Bd((size(Bm,2)*(maxn + 1)+1):(size(Bm,2)*(maxn + 1)+size(Am,1)), 1:size(Bm,2)) =
Bd((size(Bm,2)*(maxn + 1)+1):(size(Bm,2)*(maxn + 1)+size(Am,1)), 1:size(Bm,2)) + Vd(:, :, a);
    end
end
end

for a=1:maxn
    Ad((size(Bm,2)*(a - 1) + 1):(size(Bm,2)*a), size(Bm,2)*a + 1:size(Bm,2)*(a + 1)) =
eye(size(Bm,2));
end
else
    Ad = Am;
    Bd = Bm;
    Gd = Gm;
    xd0 = xm0;
end

Kd = ss(Ad, Bd, Gd, 0, Ts); %Modelo discreto final

return; %Retorna antes de criar gráfico e avaliar sistema

%*****
%Verifica se o sistema é controlável (estabilizável)
rctrb = rank(ctrb(Ad,Bd));
if rctrb < size(Ad,2)
    Aco = ctrbf(Ad, Bd, Gd);
    if abs(eig(Aco(1:(size(Ad,2)-rctrb), (rctrb+1):size(Ad,2)))) < 1
        'O sistema é estabilizável'
    else
        'O sistema não é estabilizável'
    end
else
    'O sistema é controlável'
end

%*****
%Verifica se o sistema é observável (detectável)
robsv = rank(obsv(Ad,Gd));
if robsv < size(Ad,2)
    Aco = obsvf(Ad, Bd, Gd);
    if abs(eig(Aco(1:(size(Ad,2)-robsv), 1:(size(Ad,2)-robsv)))) <= 1
        'O sistema é detectável'
    else
        'O sistema não é detectável'
    end
else
    'O sistema é observável'
end

return; %Retorna antes de criar gráfico

if max(abs(xd0))==0
    step(Kd, 10);
end

```

```

else
    initial(Kd, xd0, 10);
end
grid;

%INTEGEXP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Gera integral de exp(A*t)
%A é matriz
%li = limite inferior
%ls = limite superior

function Int = integexp(A, ls, li)

if (det(A) ~= 0)
    Int = inv(A)*(expm(A*ls) - expm(A*li));
else
    Int = eye(size(A,1))*(ls-li);
    for a=2:100
        Intemp = ((A^(a-1))/(factorial(a)))*(ls^a - li^a);
        if max(abs(Intemp)) < 1e-9
            break;
        else
            Int = Int + Intemp;
        end
    end
end
end

%INTEGINVEXP %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Gera integral de exp(-A*t)
%A é matriz
%li = limite inferior
%ls = limite superior

function Int = integinvexp(A, ls, li)

if (det(A) ~= 0)
    Int = -inv(A)*(expm(-A*ls) - expm(-A*li));
else
    Int = eye(size(A,1))*(ls-li);
    for a=2:100
        Intemp = (((-A)^(a-1))/(factorial(a)))*(ls^a - li^a);
        if max(abs(Intemp)) < 1e-9
            break;
        else
            Int = Int + Intemp;
        end
    end
end
end

%INTEGINVEXPLIN %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Gera integral de t*exp(-A*t)
%A é matriz
%li = limite inferior
%ls = limite superior

function Int = integinvexplin(A, ls, li)

Int = 1*(eye(size(A,1))*(1/2))*(ls^2-li^2);
for a=2:100
    Intemp = a*(((A)^(a-1))/(factorial(a+1)))*(ls^(a+1) - li^(a+1));
    if max(abs(Intemp)) < 1e-9
        break;
    else
        Int = Int + Intemp;
    end
end
end

```

## B.4 Função REGIME

```

%*****
% Função REGIME
% Calcula o estado e a saída de regime
%
% Entrada
% Kd      = Rede de DPLL's discretizada
% u       = Entrada do sistema
% Ts      = Período de amostragem
% dimpll  = Tamanho de cada nó independentemente
%
% Saída
% yreg = Saída do sistema em regime
% xreg = Estado do sistema em regime
% autoval = Autovalores do sistema
%*****

function [yreg, xreg, autoval] = regime(Kd, u, Ts, dimpll)

%Cria valores temporários das matrizes para tratamento
Ad = Kd.a;
Bd = Kd.b;
Gd = Kd.c;

%Calcula autovalores
autoval = eig(Ad);
maxautoval = max(abs(autoval)) %Autovalor máximo
if maxautoval > 1.000001
    xreg = 0;
    yreg = 0;
    return;
end

%Calcula dimensão da rede no tempo contínuo
dimredecon = sum(dimpll);
dimrededis = size(Ad,1);

%Gera matriz mudança de base T
T = eye(dimrededis);
index = 1;
for m = 1:(dimrededis/dimredecon)
    for k = 1:size(dimpll,2)
        T(index,1) = 1;
        index = index + dimpll(k);
    end
end

%Anula primeira coluna de Ad para determinar ws*Ts
Ad(:,1) = 0*Ad(:,1);

%Determina estado e saída de regime
xreg = inv(T - Ad)*Bd*u;
yreg = Gd*xreg;

%Calcula valor da frequência de regime
ws = xreg(1)/Ts;
fs = ws/(2*pi)

return;

%*****
%Verifica se o sistema é controlável (estabilizável)
rctrb = rank(ctrb(Kd.a,Kd.b));
if rctrb < size(Kd.a,2)
    Aco = ctrbf(Kd.a, Kd.b, Kd.c);
    if abs(eig(Aco(1:(size(Kd.a,2)-rctrb),(rctrb+1):size(Kd.a,2)))) < 1
        'O sistema é estabilizável'
    else
        'O sistema não é estabilizável'
    end
end

```



```
end
else
    'O sistema é controlável'
end

%*****
%Verifica se o sistema é observável (detectável)
robsv = rank(observ(Kd.a,Kd.c));
if robsv < size(Kd.a,2)
    Aco = obsvf(Kd.a, Kd.b, Kd.c);
    if abs(eig(Aco(1:(size(Kd.a,2)-robsv),1:(size(Kd.a,2)-robsv)))) <= 1
        'O sistema é detectável'
    else
        'O sistema não é detectável'
    end
else
    'O sistema é observável'
end
```