

# Gerenciamento de Energia em Sistemas Embarcados

Saulo Oliveira Dornellas Luiz

Dissertação de Mestrado submetida à Coordenação do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Campina Grande - Campus de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração: Processamento da Informação

Angelo Perkusich, D.Sc.

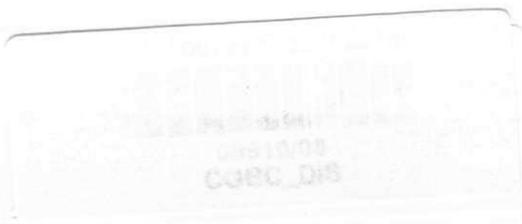
Orientador

Antonio Marcus Nogueira Lima, Dr.

Orientador

Campina Grande, Paraíba, Brasil

©Saulo Oliveira Dornellas Luiz, Março de 2008





**FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA  
CENTRAL DA UFPG**

L953g

2008 Luiz, Saulo Oliveira Dornellas.

Gerenciamento de energia em sistemas embarcados / Saulo  
Oliveira Dornellas Luiz. — Campina Grande, 2008.  
137f. : il. color.

Dissertação (Mestrado em Engenharia Elétrica) – Universidade  
Federal de Campina Grande, Centro de Engenharia Elétrica e  
Informática.

Referências.

Orientadores: Ângelo Perkusich, D.Sc., Antonio Marcus Nogueira  
Lima, Dr.

1. Gerenciamento de energia. 2. Cadeias de Markov. 3. Bateria. 4.  
Sistemas embarcados. I. Título.

CDU- 621.316:519.217.2(043)

**GERENCIAMENTO DE ENERGIA EM SISTEMAS EMBARCADOS**

**SAULO OLIVEIRA DORNELLAS LUIZ**

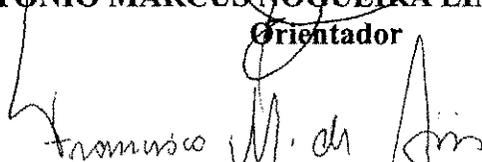
Dissertação Aprovada em 14.03.2008



**ANGELO PERKUSICH, D.Sc., UFCG**  
Orientador



**ANTONIO MARCUS NOGUEIRA LIMA, Dr., UFCG**  
Orientador



**FRANCISCO MARCOS DE ASSIS, Dr., UFCG**  
Componente da Banca



**JOSÉ SÉRGIO DA ROCHA NETO, D.Sc., UFCG**  
Componente da Banca



**PÉRICLES REZENDE BARROS, Ph.D., UFCG**  
Componente da Banca

CAMPINA GRANDE - PB  
MARÇO-2008

A Alba, José Luiz e Simone.

## Agradecimentos

A Deus, que me fortalece e me conduz.

À minha família, especialmente meus pais, Alba e José Luiz, minha irmã Simone e meus avós, pelo seu amor, seus esforços para me conduzirem a esse caminho, e seus exemplos de vida que me emocionam e me dão forças para ir cada vez mais longe. A Otávia, pelo amor, carinho, estímulo e companheirismo.

Aos professores Angelo Perkusich e Antonio Marcus Nogueira Lima, pela orientação, atenção, apoio e tempo dedicado.

Aos professores com os quais paguei as disciplinas, pela sua contribuição na minha formação.

Aos integrantes da UAEE e os funcionários da COPELE, pelo apoio.

À Capes, pela bolsa de pesquisa.

A todos os alunos, professores e funcionários do Laboratório de Sistemas Embarcados e Computação Pervasiva, que contribuíram direta ou indiretamente para a realização desse trabalho.

## Resumo

A complexidade de sistemas embarcados alimentados a bateria tais como telefones móveis, e assistentes digitais pessoais (PDAs) está crescendo rapidamente. Dispositivos baseados em processadores de alta velocidade e múltiplos núcleos, tendo múltiplas câmeras, dispositivos de visualização e várias interfaces de rede aumentam a demanda por mais energia. Contudo, a capacidade das baterias não cresce na mesma taxa em que cresce a complexidade dos dispositivos móveis embarcados. Assim, o aumento da autonomia da bateria usando estratégias de gerenciamento de energia se tornou um dos desafios chave no projeto de sistemas embarcados móveis complexos. As técnicas dinâmicas de gerenciamento de energia, denominadas gerenciamento dinâmico de energia (GDE), permitem a redução do consumo de energia em tempo de execução desligando ou reduzindo a frequência ou tensão de componentes inativos do sistema. Uma estratégia de gerenciamento de energia deve levar em conta a carga de trabalho do sistema. Num processador de propósito geral, por exemplo, a combinação das aplicações executando em tal sistema pode variar bastante, dependendo do que está sendo executado. Além disso, a carga de trabalho pode variar drasticamente ao longo do dia, ou ao longo dos dias da semana, ou quando o sistema é operado por usuários diferentes. Isso ocorre devido à não-estacionaridade da carga de trabalho. Além disso, estratégias tradicionais de otimização de consumo de energia podem não ser ótimas para dispositivos alimentados a bateria se as características da bateria não forem adequadamente modeladas e exploradas. Para otimizar a autonomia da bateria devem ser todos levados em conta: a carga de trabalho do sistema, os parâmetros elétricos (e.g. valores de correntes) do sistema eletrônico, e as características eletroquímicas da bateria. Nesse trabalho é proposta uma técnica de GDE orientada a autonomia da bateria, que explora um modelo analítico acurado de bateria para aumentar a autonomia da bateria num ambiente não-estacionário. O sistema é modelado por cadeias de Markov no tempo discreto, em associação ao modelo de bateria. Tal modelo do sistema permite uma formulação matemática rigorosa do problema e uma solução de compromisso entre desempenho e autonomia da bateria. A técnica de GDE proposta foi simulada em Matlab e implementada usando a plataforma OMAP 1611 da Texas Instruments executando o sistema operacional Linux. Através dos resultados de simulação e experimentais verificou-se que a técnica aqui introduzida resulta em maiores autonomias de bateria em comparação a técnicas de GDE anteriores.

## Abstract

The complexity of battery-powered embedded system such as mobile phones, and personal digital assistants (PDAs) is growing quite fast. Devices based on high speed and multi-core cpus having multiple cameras, display devices and several network interfaces increase the demand for more power. However, battery capacity does not grow at the same rate as it does the complexity of mobile embedded devices. Therefore, extending battery lifetime using power management strategies has become one of the key challenges in the design of complex mobile embedded systems. The dynamic power management techniques, named (DPM), allow power reduction at runtime by shutting down or reducing frequency or voltage of idle system components. An strategy of power management must consider the workload of the system. At a general purpose processor e.g. the combination of applications running on such system may vary strongly, depending on what is being executed. Moreover, the workload may vary drastically during the day, or over the days of the week, or when the system is operated by different users. It happens because of the nonstationarity of the workload. Besides, traditional power optimization strategies may not be optimal for battery-powered devices if the characteristics of the battery are not properly modeled and exploited. In order to optimize the battery lifetime, all of these must be taken into account: the model of the workload of the system, the electric parameters (e.g. values of currents) of the electronic system and the electrochemical features of the battery. A battery-aware DPM technique that exploits an accurate analytical battery model to increase battery lifetime in a non-stationary environment is proposed in this work. The system is modeled by discrete-time Markov chains coupled to the battery model. Such model allows a rigorous mathematical formulation of the problem and a trade-off between performance and battery lifetime. The proposed DPM technique has been simulated at Matlab and implemented using the Texas Instruments OMAP 1611 platform running Linux. Simulation and experimental results have shown that the technique introduced here results in longer battery lifetimes compared to previous DPM techniques.

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Gerenciamento dinâmico de energia</b>	<b>6</b>
2.1	GDE em ambiente estacionário e conhecido . . . . .	6
2.2	GDE em ambiente não-estacionário . . . . .	26
2.2.1	Abordagem com janela única . . . . .	26
2.2.2	Abordagem com multi-janela . . . . .	28
2.2.3	Obtenção da política de energia . . . . .	31
2.3	Conclusões . . . . .	38
<b>3</b>	<b>Modelos de bateria</b>	<b>39</b>
3.1	Modelo Analítico . . . . .	41
3.2	Descarga da bateria e tempo de recuperação . . . . .	47
3.3	Conclusões . . . . .	50
<b>4</b>	<b>Gerenciamento dinâmico de energia orientado à autonomia da bateria</b>	<b>51</b>
4.1	Função custo de bateria no problema de otimização de GDE . . . . .	52
4.2	GDE orientado à autonomia da bateria em ambiente não-estacionário . . . . .	53
4.3	Conclusões . . . . .	58
<b>5</b>	<b>Metodologia experimental</b>	<b>59</b>
5.1	Gerenciamento de energia no OMAP1611 através do projeto DPM . . . . .	59
5.1.1	Políticas DPM . . . . .	60
5.1.2	Uso de temporizadores de propósito geral no OMAP1611 . . . . .	62
5.2	Medição da tensão e corrente da bateria . . . . .	66
5.3	Caracterização da bateria para validação de GDEBAT . . . . .	67
5.4	Implementação de GDEBAT . . . . .	70
5.5	Estudo de caso . . . . .	71
5.6	Conclusões . . . . .	80

<b>6</b>	<b>Conclusões e trabalhos futuros</b>	<b>82</b>
6.1	Trabalhos futuros . . . . .	83
	<b>Referências Bibliográficas</b>	<b>84</b>

# Glossário

ACPI	Interface Avançada de Configuração e Energia ( <i>Advanced Configuration and Power Interface</i> )
API	<i>Application Programming Interface</i>
ARM	<i>Advanced RISC machine</i>
BRA	Buffer da Requisição Anterior
CM	Cadeia de Markov
CMTC	Cadeia de Markov no tempo contínuo
CMTD	Cadeia de Markov no tempo discreto
DPLL	<i>Digital phase-locked loop</i>
DPM	<i>Dynamic Power Management</i>
EA	Estado Atual
EA	Estado Atual
EDT	Escalonamento Dinâmico de Tensão ( <i>Dynamic Voltage Scaling</i> )
EDTF	Escalonamento Dinâmico de Tensão e Frequência ( <i>Dynamic Voltage and Frequency Scaling</i> )
EMQ	Erro Médio Quadrático ( <i>Mean Square Error</i> )
FS	Fila de Serviço
GDEBAT	Gerenciamento Dinâmico de Energia orientado à autonomia de Bateria em ambiente não-estacionário
OMAP	Plataforma Aberta para Aplicações Multimídia ( <i>Open Multimedia Application Platform</i> )
PC	Computador Pessoal ( <i>Personal Computer</i> )
PDMTCC	Processo de Decisão de Markov no Tempo Contínuo ( <i>Continuous-time Markov Decision Process</i> )
PDS	Processador Digital de Sinais ( <i>Digital Signal Processor</i> )

PL	Programação Linear
PO	Problema de Otimização
PS	Provedor de Serviços
RISC	<i>Reduced Instruction Set Computer</i>
RS	Requisitante de Serviços
RS	Requisitante de Serviços
SO	Sistema Operacional
TC	<i>Traffic Controller</i>
UCP	Unidade Central de Processamento ( <i>Central Processing Unit</i> )
USB	<i>Universal Serial Bus</i>

# Lista de Tabelas

2.1	Erro médio quadrático da identificação do RS para a abordagem com método de janela única . . . . .	28
2.2	Erro médio quadrático da identificação do RS para a abordagem com método de multi-janela . . . . .	31
2.3	Comparação de algoritmos. . . . .	37
3.1	Autonomias de uma bateria para diferentes correntes constantes. . . . .	43
4.1	Comparação de algoritmos. . . . .	58
5.1	Alguns pontos de operação do OMAP1611 . . . . .	61
5.2	Exemplos de mapeamentos de pontos de operação do OMAP1510 a estados de operação . . . . .	62
5.3	Autonomias de uma bateria Siemens Li-Ion de 3,7V e 820mAh para diferentes ensaios a corrente constante. . . . .	70
5.4	Comparação de algoritmos. . . . .	75
5.5	Variando $k$ para GDEBAT . . . . .	76
5.6	Resultado experimental para ensaio do PS sempre no estado 192 . . . . .	80
5.7	Resultado experimental para ensaio do PS sempre no estado 60 . . . . .	80
5.8	Resultados experimentais para GDEBAT . . . . .	81
5.9	Resultados experimentais para a técnica de Chung et al.(1) . . . . .	81

# Lista de Figuras

2.1	Componentes do modelo do sistema baseado em processos de Markov no tempo discreto. . . . .	7
2.2	Exemplo de modelo em cadeia de Markov para um requisitante de serviços. . . . .	12
2.3	O PS não serviu nenhuma requisição e o número de requisições que chegaram mais as que já estavam na fila não excede o tamanho da fila. . . . .	14
2.4	O PS serviu uma requisição e a soma do número de requisições que estavam na fila mais as que chegaram menos a que foi servida não excede o tamanho da fila. . . . .	15
2.5	A fila estava vazia e assim permanece pois não aparece nenhuma nova requisição. . . . .	16
2.6	A fila se torna cheia e a soma do número de requisições que estavam na fila mais as que chegaram excede o tamanho da fila. . . . .	17
2.7	FS de comprimento 2. . . . .	18
2.8	Exemplo de modelo em cadeia de Markov para um provedor de serviços. . . . .	20
2.9	Escolha do comando que o GE passará ao PS. . . . .	24
2.10	Início de uma simulação do sistema do exemplo 6 . . . . .	25
2.11	Exemplo de um RS não-estacionário. . . . .	26
2.12	Exemplo de uso de uma janela única para RS de dois estados. . . . .	27
2.13	Identificação de um RS através do método de janela única com comprimento 10. . . . .	29
2.14	Identificação de um RS através do método de janela única com comprimento 50. . . . .	29
2.15	Identificação de um RS através do método de janela única com comprimento 100. . . . .	30
2.16	Exemplo de uso de multi-janela para RS de dois estados. . . . .	30
2.17	Identificação de um RS através do método de multi-janela com comprimento 10. . . . .	32
2.18	Identificação de um RS através do método de multi-janela com comprimento 50. . . . .	32

2.19	Identificação de um RS através do método de multi-janela com comprimento 100. . . . .	33
2.20	Exemplo de tabela de políticas com $NS_0 = NS_1 = 5$ . . . . .	34
2.21	Tabela de decisão e exemplo de seleção de linhas da tabela de políticas. . .	35
2.22	Exemplo de tabela de políticas com $NS_0 = NS_1 = 5$ . . . . .	35
2.23	Identificação do RS definido em (2.52) e (2.53) através do método de multi-janela com comprimento 100. . . . .	37
3.1	Efeito de recuperação na bateria. . . . .	40
3.2	Autonomia da bateria em função de corrente de descarga constante. . . . .	44
3.3	Perfil de descarga para a bateria. . . . .	44
3.4	Perfis de descarga. . . . .	46
3.5	Carga residual da bateria para cada perfil de descarga da Figura 3.4. . . .	47
3.6	Exemplo de um processo de descarga. . . . .	48
4.1	Escolha de tabela de decisão com base na identificação do RS e na penalidade de desempenho a ser usada. . . . .	55
4.2	Exemplo de uso de uma função exponencial para variar restrições de penalidade de desempenho quando o PS desliga. . . . .	56
5.1	Exemplo de estados de operação e transições candidatas a uma implementação no projeto DPM. . . . .	60
5.2	Interações entre componentes de um sistema com DPM. . . . .	61
5.3	Transição entre políticas de energia “alta-potência” e “baixa-potência”, cujos pontos de operação no estado “tarefa” são “192” e “60” respectivamente. . .	63
5.4	Transição entre pontos de operação 192 e sono-168, estado em que permanece o OMAP por 5 segundos e depois retorna ao estado 192. . . . .	64
5.5	Transição entre pontos de operação 60 e sono-60, estado em que permanece o OMAP por 5 segundos e depois retorna ao ponto de operação sono-60. . .	65
5.6	Diagrama de blocos para medição da tensão e corrente da bateria. . . . .	66
5.7	Diagrama elétrico para medição da tensão e corrente da bateria através do módulo de aquisição de dados NI USB-6210. . . . .	67
5.8	Diagrama de blocos no programa LabVIEW. . . . .	68
5.9	Painel frontal no programa LabVIEW. . . . .	68
5.10	Esquema para ensaios a corrente constante para caracterização da bateria. .	69
5.11	Diagrama elétrico do circuito utilizado para descarga ou carga da bateria a corrente constante. . . . .	69
5.12	Ensaio a corrente constante para caracterização da bateria. . . . .	71

5.13 Diagrama de fluxo do cálculo da tabela de políticas antes do tempo de execução. . . . .	72
5.14 Diagrama de fluxo da política de gerenciamento de energia em tempo de execução. . . . .	73
5.15 Estados do sistema ao longo do primeiro ensaio com GDEBAT. . . . .	77
5.16 Identificação do RS ao longo do primeiro ensaio com GDEBAT. . . . .	78
5.17 Tensão, corrente e potência da bateria ao longo do primeiro ensaio com GDEBAT. . . . .	78
5.18 Estados do PS e corrente fornecida pela bateria ao longo do primeiro ensaio com GDEBAT, e ampliação para o intervalo de 130 a 180s. . . . .	79

# Capítulo 1

## Introdução

O consumo de energia de sistemas embarcados alimentados a bateria tais como telefones celulares, *tablets* para navegação na internet e assistentes pessoais (PDAs) está aumentando rapidamente devido ao crescente uso de processadores de alta velocidade e múltiplos núcleos, câmeras, LCDs, várias interfaces de rede, etc. Isso tem se tornado um problema crítico em dispositivos móveis alimentados a bateria devido ao fato de que a capacidade das baterias não tem aumentado tão rápido quanto a demanda de consumo de energia por tais dispositivos (1, 2). Assim o desenvolvimento de técnicas para o aumento da autonomia da bateria através do uso eficiente de energia se tornou um desafio chave para o projeto de sistemas embarcados.

As técnicas de redução de consumo de energia são classificadas em estáticas e dinâmicas (3). As técnicas estáticas são usadas na fase de projeto; são exemplos a síntese e a compilação para baixo consumo. Já as técnicas dinâmicas, denominadas *Gerenciamento Dinâmico de Energia* (GDE), possibilitam a redução do consumo de energia nesses sistemas em tempo de execução, através do desligamento ou redução da frequência/tensão de operação de componentes de sistemas que estão ociosos ou sub-utilizados. Num dispositivo, um *gerenciador de energia* (GE) monitora os estados dos componentes e do sistema e controla seus estados de energia através de um procedimento denominado *política de gerenciamento de energia* (1), que decide quais estados de operação vão assumir os componentes do sistema assim como o próprio sistema (4).

Apesar de uma técnica de GDE possibilitar a redução do consumo de energia de um dispositivo, ela pode não ser ótima para um sistema alimentado por bateria se não forem consideradas as características do comportamento da bateria (2). Destacam-se dois importantes fenômenos nas baterias: a característica de taxa de capacidade (*Rate capacity*) e o fenômeno de relaxação ou efeito de recuperação (*Recovery effect*). A característica de taxa de capacidade origina-se do fato de que, quanto maior a corrente de descarga, menor será a capacidade disponível da bateria. O fenômeno de relaxação é ocasionado pelo

gradiente de concentração de materiais ativos no eletrólito, formado durante a descarga da bateria. O material ativo na interface eletrólito-eletrólito é consumido pelas reações eletroquímicas, e durante um período em que a bateria não é utilizada, esse material é repostado por novo material ativo através de difusão, fazendo com que a capacidade da bateria seja recuperada. Esses fenômenos contribuem para que uma política de baixo consumo de energia não implique necessariamente em maior autonomia da bateria, pois a capacidade de energia da bateria pode não ser completamente utilizada no momento em que ela atinge a tensão de corte (*cut-off voltage*), ou seja, aquela em que a bateria é considerada descarregada.

Verifica-se então a importância de que sejam considerados tanto o modelo do sistema eletrônico portátil quanto o de sua bateria. Com base em um modelo baseado em tais elementos é possível levar em consideração o comportamento estatístico das tarefas a serem atendidas, as características elétricas (e.g. valores de correntes) do sistema eletrônico e as características eletroquímicas das baterias (2).

## Revisão Bibliográfica

Paleologo et al.(4) destacam que GDE encontra a sua aplicação em sistemas ou seus componentes que passam por cargas de trabalho (*workloads*) não-uniformes durante o tempo de operação, como encontrado em redes de comunicação e sistemas interativos.

Uhrig e Ungerer(5) apontam que várias técnicas de redução de consumo de energia estão presentes nos processadores comerciais, tais como: *pipeline gating*, modos de operação do tipo suspender (*suspend*) ou dormir (*sleep*), e redução de tensão e frequência. São exemplos de famílias de processadores que usam escalonamento de frequência e tensão: *XScale* da Intel, *Crusoe* da Transmeta e *OMAP* da Texas Instruments (6). A plataforma OMAP 1611, por exemplo, é composta de processadores de alto desempenho e direcionada para dispositivos de navegação na Internet, PDAs, etc. A plataforma OMAP 1611 possui dois núcleos: um *Advanced Risk Machine (ARM)* TI926TEJ e um Processador Digital de Sinais (PDS) TMS320C55x de baixo consumo. O desenvolvimento de aplicações para a plataforma OMAP, descrita por Luiz et al.(7), impõe vários desafios tais como comunicação entre processadores, limitações de uso de memória e possivelmente conhecimento sobre a arquitetura de hardware específica de cada núcleo da plataforma. Foram introduzidas por Gorgônio et al.(8) técnicas para auxiliar o desenvolvedor na escolha sobre que partes de uma aplicação destinar a cada núcleo da plataforma tendo como critério o desempenho da aplicação. Luiz et al.(9) apresentaram modelos formais para o mecanismo de transmissão de dados entre ARM e DSP na plataforma OMAP via DSP Gateway. Tais modelos são representados como autômatos temporizados e auxiliam desenvolvo-

res a compreender o mecanismo de comunicação via DSP Gateway e facilitam seu uso e desenvolvimento futuro.

Em relação à política de *timeout*, Chung et al.(1) afirmam que, devido à sua simplicidade, tal política é amplamente utilizada em microprocessadores, monitores, discos rígidos, etc. Como descrevem Lu e Micheli(3), tal política é baseada na hipótese de que após um dispositivo tornar-se inativo por um dado intervalo de tempo denominado *timeout*, ele permanecerá inativo por pelo menos o intervalo de tempo *break-even*. Este intervalo é o tempo mínimo de inatividade para o qual é interessante desligar o dispositivo dado que a economia de energia no modo de mais baixa energia compensará a energia gasta para desligar e religar o dispositivo. Chung et al.(1) apontam que o uso do intervalo de tempo *break-even* resulta numa política que oferece economia de energia dentro de um fator de 2 em relação à economia alcançada por uma política ideal que possua conhecimento perfeito do futuro, a qual é denominada política *oráculo*. Uma desvantagem relevante da política de *timeout* é o gasto de energia durante a espera para o *timeout* expirar.

Simunic et al.(10) combinam GDE e escalonamento dinâmico de tensão (EDT) através da expansão de estado ativo de modo a incluir múltiplos conjuntos de frequências e tensões, o que permite que o GE utilize DVS para controlar consumo de energia e desempenho quando o sistema está ativo, e utilizar estados de mais baixo consumo quando o sistema está desocupado.

Rakhmatov e Vruthula(11) introduziram um modelo para a predição do tempo de falha da bateria (*time-to-failure*) para uma dada carga, e uma métrica de custo para algoritmos de otimização da autonomia da bateria. Posteriormente, Rakhmatov e Vruthula(12) utilizaram um modelo analítico de bateria para a realização de agendamento (*scheduling*) de tarefas e escalonamento (*scaling*) de tensões. Benini et al.(13) propuseram uma técnica de gerenciamento de energia de malha fechada que monitora a tensão de saída da bateria para estender a autonomia da bateria. É definido um limite de tensão de saída da bateria, acima do qual o sistema assume um modo de operação de alto desempenho, e abaixo do qual escolhe-se um modo de operação de baixo desempenho.

Algumas abordagens de GDE tomam as aplicações que executam no sistema operacional (SO) como referências para o gerenciamento de energia. Zhong e Jha(14) empregaram a informação e história da interface com o usuário, e teorias de Psicologia para realizar a predição dos atrasos do usuário, utilizando-os em técnicas de GDE/DVS. Anand, Nightingale e Flinn(15) fizeram as aplicações liberarem “dicas fantasmas” (*ghost hints*) que auxiliam o gerenciamento de energia quando há múltiplos dispositivos. Contudo, Chung et al.(1) afirmam que o gerenciamento de energia realizado por aplicações não é viável quando as aplicações não podem ser modificadas.

Paleologo et al.(4) apresentaram uma técnica que consiste de um modelo estocástico baseado em processos de Markov para a formulação da política de energia. É obtida uma solução exata do problema de otimização da política de energia em tempo polinomial resolvendo-se um problema de otimização linear. Não é considerado um modelo de bateria e então o problema de otimização é orientado apenas à redução do consumo de energia. Além disso, o problema de otimização é apenas resolvido para o caso estacionário, o que em grande quantidade dos problemas reais de GDE pode ser uma desvantagem, dado que é difícil ou impossível caracterizar previamente a carga de trabalho do sistema, devido à não-estacionaridade.

Rong e Pedram(2) propuseram um modelo estocástico de um sistema alimentado a bateria baseado em processos de decisão de Markov no tempo contínuo (PDMTC) . É apresentado um modelo de bateria que considera a característica de taxa de capacidade e o fenômeno de relaxação. O problema de otimização da política de energia é baseado nas teorias de PDMTC e resolvido linearmente através de uma abordagem de programação linear. Nesse trabalho também é considerado apenas o caso estacionário.

Chung et al.(1) apontaram desvantagens na formulação de GDE como um problema estocástico ótimo de controle quando é apenas considerado o caso estacionário, como realizado por Paleologo et al.(4) no tempo discreto e Rong e Pedram(2) no tempo contínuo. Assim Chung et al.(1) apresentaram um esquema de GDE adaptativo que permite lidar com cargas de trabalho inicialmente desconhecidas e não-estacionárias. Esta técnica de GDE não considera o modelo de bateria e então o problema de otimização é orientado apenas à redução do consumo de energia.

## Objetivos

Dadas as limitações apresentadas pelos trabalhos anteriores, neste trabalho tem-se como objetivo modelar, conceber, implementar e avaliar uma política de gerenciamento de energia para sistemas embarcados orientada à otimização da autonomia de bateria em que se contemple as seguintes características do sistema:

- Modelo da bateria que inclui a característica de taxa de capacidade e o fenômeno de relaxação;
- Modelo estocástico do sistema, baseado em processos de Markov no tempo discreto;
- Modelo não-estacionário para a carga de trabalho do sistema.

## Organização do Texto

O restante desse texto está disposto em cinco capítulos. No Capítulo 2 são apresentadas duas técnicas de gerenciamento dinâmico de energia, a primeira utiliza processos de Markov no tempo discreto para o caso estacionário, e a segunda é a extensão da primeira para o caso não-estacionário. No Capítulo 3 são apresentados modelos de bateria. No Capítulo 4 é apresentada a técnica de gerenciamento dinâmico de energia proposta. No Capítulo 5 é apresentada a metodologia experimental para a implementação e avaliação da política de gerenciamento dinâmico de energia proposta. No Capítulo 6 são apresentadas as conclusões e os trabalhos futuros.

## Capítulo 2

# Gerenciamento dinâmico de energia

Sistemas com gerenciamento de energia oferecem mecanismos para alterar o estado de energia de seus componentes. Cada um desses estados define um desempenho e consumo de energia específicos. Para promover redução do consumo de energia, as políticas de GDE podem utilizar informações como a história do sistema, a carga de trabalho ou restrições de desempenho para decidir qual transição de estado deve ser realizada e quando. Essas políticas procuram uma solução de compromisso entre consumo de energia e desempenho.

É possível imaginar uma política agressiva (denominada política *ávida*) que desliga cada componente assim que ele fica inativo. Quando o componente é solicitado novamente, ele é religado, processo que envolve energia e tempo. Como geralmente reduz o desempenho e pode nem ao menos reduzir o consumo de energia, a política ávida é freqüentemente inaceitável (4). As técnicas de GDE baseadas em processos de Markov procuram superar tais problemas modelando o sistema de forma estocástica e resolvendo um problema de otimização para a política de energia.

### 2.1 GDE em ambiente estacionário e conhecido

O sistema com gerenciamento de energia é modelado por Paleologo et al.(4) como um *provedor de serviços* (PS) e um GE. O provedor de serviços processa as requisições de um único *requisitante de serviços* (RS), e possui vários estados de operação, cada um com uma taxa de serviço. É possível que haja uma *fila de serviço* (FS) que armazena requisições que ainda não puderam ser servidas pelo provedor de serviços, ou porque ele está ocupado ou está com taxa de serviço nula. Esse modelo do sistema é ilustrado na Figura 2.1. Assume-se que as requisições são indistinguíveis, não havendo prioridade de serviço, e a fila de serviço possui tamanho finito. O gerenciador de energia toma decisões sobre o estado do provedor de serviços com base na história do sistema.

Na notação a seguir, são utilizadas letras maiúsculas em negrito (e.g., **M**) para denotar

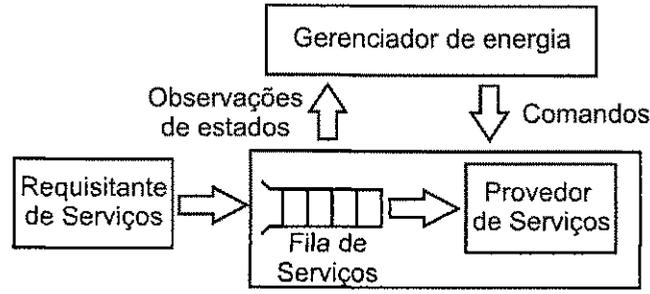


Figura 2.1: Componentes do modelo do sistema baseado em processos de Markov no tempo discreto.

matrizes, letras minúsculas em negrito (e.g.,  $\mathbf{v}$ ) para denotar vetores, letras caligráficas (e.g.,  $\mathcal{S}$ ) para conjuntos, letras maiúsculas itálicas (e.g.,  $S$ ) para constantes escalares, e letras minúsculas itálicas (e.g.,  $x$ ) para variáveis escalares. No tempo discreto  $t_n = Tn$ , tem-se que  $T$  é a resolução no tempo, e  $n \in \mathbb{N}_+$ . Escreve-se  $x_n$  em substituição a  $x_{t_n}$ . Segue a definição de processo de Markov, com o qual é modelado o sistema com gerenciamento de energia.

### Definição 1

Seja o processo aleatório  $\{X(t), t \in \mathcal{I}\}$  que assume valores de um conjunto  $\mathcal{S}$ , denominado espaço de estados. O conjunto  $\mathcal{I}$  é o conjunto de índices e pode ser enumerável ou não-enumerável.  $\mathcal{I}$  representa aqui um conjunto de índices de tempo, como, por exemplo,  $\mathcal{I} = (-\infty, \infty)$  ou  $\mathcal{I} = [0, \infty)$ . Um processo aleatório  $\{X(t), t \in \mathcal{I}\}$  é um processo de Markov (16) se

$$\begin{aligned} P[X(t_{n+1}) \leq j | X(t_n) = i, X(t_{n-1}) = i_{n-1}, \dots, X(t_1) = i_1, X(t_0) = i_0] = \\ = P[X(t_{n+1}) \leq j | X(t_n) = i] \end{aligned} \quad (2.1)$$

para qualquer escolha de  $n > 1$ ,  $\{t_k\} \in \mathcal{I}$  com  $t_0 < t_1 < \dots < t_{n+1}$ , e  $i, j, \{i_k\} \in \mathcal{S}$ .

Se o espaço de estados  $\mathcal{S}$  é discreto, o processo de Markov é denominado uma *Cadeia de Markov* (CM). Uma Cadeia de Markov pode ser: no tempo discreto ou no tempo contínuo, caso o conjunto de índices de tempo  $\mathcal{I}$  for discreto ou contínuo respectivamente. E uma Cadeia de Markov pode ser classificada como finita ou infinita, caso o espaço de estados  $\mathcal{S}$  contiver, respectivamente, um número finito ou infinito de elementos.

Dado que numa cadeia de Markov no tempo discreto o espaço de estados  $\mathcal{S}$  e o conjunto de índices  $\mathcal{I}$  são discretos, e qualquer conjunto discreto pode ser mapeado para o conjunto de inteiros, os elementos de  $\mathcal{S}$  e  $\mathcal{I}$  podem ser rotulados por inteiros. É possível então assumir para cadeias de Markov no tempo discreto que  $\mathcal{S} = \{\dots, -2, -1, 0, 1, 2, \dots\}$  ou  $\mathcal{S} = \{0, 1, 2, \dots\}$ , e que  $\mathcal{I} = \{0, 1, 2, \dots\}$ , ou seja,  $t_i = i$ ,  $i = 0, 1, \dots$ . É possível também

denotar a variável aleatória  $X(t_n)$  por  $X_n$ . Com base em tal classificação e notação, é enunciada a definição de cadeia de Markov no tempo discreto:

**Definição 2**

Um processo aleatório  $\{X_n, n = 0, 1, \dots\}$  é uma Cadeia de Markov no tempo discreto (CMTD) (16) se o espaço de estados  $\mathcal{S}$  e o conjunto de índices  $\mathcal{I}$  são discretos e

$$P[X_{n+1} = j | X_n = i, X_{n-1} = i_{n-1}, \dots, X_0 = i_0] = P[X_{n+1} = j | X_n = i] \quad (2.2)$$

para qualquer escolha de  $n$  e  $i, j, \{i_k\} \in \mathcal{S}$ . Seja a matriz  $P(n)$ , cujo  $ij$ -ésimo elemento é

$$P_{ij}(n) \triangleq P[X_{n+1} = j | X_n = i] \quad (2.3)$$

A matriz  $P(n)$  (Equação (2.4)) é denominada a matriz de transição da CMTD.

$$P(n) = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & P_{00}(n) & P_{01}(n) & \dots & P_{0l}(n) & \dots \\ \dots & P_{10}(n) & P_{11}(n) & \dots & P_{1l}(n) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & P_{l0}(n) & P_{l1}(n) & \dots & P_{ll}(n) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (2.4)$$

**Definição 3**

Seja o conjunto  $\mathcal{I} = [0, \infty)$ , e  $\{t_k\} \in \mathcal{I}$  tal que  $t_1 < t_2 < \dots < t_{n-1} < t_n$ . Um processo aleatório  $\{X(t), t \in \mathcal{I}\}$  é uma Cadeia de Markov no tempo contínuo (CMTC) (16) se

$$\begin{aligned} P[X(t_{n+1}) = j | X(t_n) = i, X(t_{n-1}) = i_{n-1}, \dots, X(t_1) = i_1, X(t_0) = i_0] = \\ = P[X(t_{n+1}) = j | X(t_n) = i] \end{aligned} \quad (2.5)$$

para qualquer escolha de  $n, \{t_k\} \in \mathcal{I}$  e  $i, j, \{i_k\} \in \mathcal{S}$ . Seja a matriz  $P(t_n, t_{n+1})$ , cujo  $ij$ -ésimo elemento é

$$P_{ij}(t_n, t_{n+1}) \triangleq P[X(t_{n+1}) = j | X(t_n) = i] \quad (2.6)$$

A matriz  $P(t_n, t_{n+1})$  é denominada a matriz de transição da CMTC.

Para CMTC homogêneas, a matriz de transição  $P(t)$  (Equação (2.7)) depende apenas da diferença no tempo  $t \triangleq t_{n+1} - t_n$ .

$$P(t) = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & P_{00}(t) & P_{01}(t) & \dots & P_{0l}(t) & \dots \\ \dots & P_{10}(t) & P_{11}(t) & \dots & P_{1l}(t) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & P_{l0}(t) & P_{l1}(t) & \dots & P_{ll}(t) & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (2.7)$$

O modelo do sistema com gerenciamento de energia através de processos de Markov é restringido para o caso em que os processos envolvidos são estacionários. Segue a definição:

**Definição 4** *Seja um processo aleatório  $\{X(t), t \in \mathcal{I}\}$  que assume valores de um conjunto  $\mathcal{S}$ , denominado espaço de estados. O conjunto  $\mathcal{I}$  é o conjunto de índices e pode ser enumerável ou não-enumerável. Para qualquer inteiro  $n > 0$ , seja  $t_1, \dots, t_n$  uma sequência crescente de índices de tempo. O processo  $\{X(t)\}$  é estacionário (16) se sua função de distribuição acumulada conjunta obedece à condição em (2.8).*

$$F_{X(t_1)X(t_2)\dots X(t_n)}(x_1, x_2, \dots, x_n) = F_{X(t_1+s)X(t_2+s)\dots X(t_n+s)}(x_1, x_2, \dots, x_n) \quad (2.8)$$

para qualquer escolha de  $s$ , sob a condição de que  $t_k + s \in \mathcal{I}, \forall k$ .

A interpretação intuitiva é de que, quando um processo aleatório é estacionário, seu “comportamento” não muda com o tempo. É possível então particularizar a definição de cadeia de Markov para o caso estacionário:

**Definição 5** *Uma cadeia de Markov estacionária  $\mathcal{M}$  é um processo estocástico sobre um conjunto finito de estados  $\mathcal{S} = \{s_i | i = 1, 2, \dots, S\}$ , em que em cada instante  $t_n$ , a distribuição de probabilidade do estado no instante  $t_n$  depende apenas do estado no instante  $t_{n-1}$ . Assim tem-se que  $\text{Prob}(X_n = s_j | X_{n-1} = s_i) = p_{s_i, s_j}$ , com  $0 \leq p_{s_i, s_j} \leq 1$  e  $\sum_{s \in \mathcal{S}} p_{s_i, s} = 1$ . Tais probabilidades podem ser organizadas na forma de uma matriz de transição  $\mathbf{P}$ .*

Se uma cadeia de Markov está num estado  $s_i$  e permanece nesse estado até transitar para um estado  $s_j$ , o tempo de permanência no estado  $s_i$  possui distribuição geométrica, representada na Equação (2.9). A definição de distribuição geométrica é apresentada a seguir.

$$P [t_{s_i, s_j} = nT] = p_{s_i, s_j} p_{s_i, s_i}^{n-1}, \quad n = 1, 2, 3 \dots \quad (2.9)$$

**Definição 6** Uma variável aleatória  $X$  de distribuição geométrica (17) com parâmetro  $p \in [0, 1]$  possui espaço amostral  $S_X = \{1, 2, 3, \dots\}$  e a probabilidade do evento  $X = k$  é dado na Equação (2.10). O valor esperado da variável aleatória  $X$  é apresentado na Equação (2.11). O parâmetro  $p$  representa uma probabilidade.

$$P[X = k] = p(1 - p)^{k-1}, \quad k = 1, 2, 3, \dots \quad (2.10)$$

$$E[X] = \frac{1}{p} \quad (2.11)$$

No caso de uma cadeia de Markov com dois estados  $\mathcal{S} = \{s_i, s_j\}$ , tem-se que  $p_{s_i, s_i} = 1 - p_{s_i, s_j}$ . O tempo de permanência no estado  $s_i$  até a cadeia de Markov transitar para o estado  $s_j$  tem a distribuição geométrica apresentada na Equação (2.12) e o valor esperado do tempo de permanência  $\bar{t}_{s_i, s_j}$  é o apresentado na Equação (2.13).

$$P[t_{s_i, s_j} = nT] = p_{s_i, s_j}(1 - p_{s_i, s_j})^{n-1}, \quad n = 1, 2, 3, \dots \quad (2.12)$$

$$\begin{aligned} \bar{t}_{s_i, s_j} &= \sum_{k=1}^{\infty} kTP[t_{s_i, s_j} = kT] = T \sum_{k=1}^{\infty} kP[t_{s_i, s_j} = kT] = T \sum_{k=1}^{\infty} kp_{s_i, s_j}p_{s_i, s_i}^{k-1} = \\ &= T \sum_{k=1}^{\infty} kp_{s_i, s_j}(1 - p_{s_i, s_j})^{k-1} = Tp_{s_i, s_j} \sum_{k=1}^{\infty} k(1 - p_{s_i, s_j})^{k-1} = Tp_{s_i, s_j} \frac{1}{(p_{s_i, s_j})^2} = \\ &= \frac{T}{p_{s_i, s_j}} \end{aligned} \quad (2.13)$$

O GE passa comandos ao PS para provocar as mudanças nos seus estados de operação. O modelo que captura essa influência do GE sobre o PS é o de cadeia de Markov estacionária controlável:

**Definição 7** Uma cadeia de Markov estacionária controlável  $\mathcal{M}(\theta)$  é uma cadeia de Markov cujas probabilidades de transição  $p_{s_i, s_j}$  são função de uma variável de controle  $\theta$ . Escreve-se  $p_{s_i, s_j}(\theta) : \Theta \rightarrow [0, 1]$ , em que  $\Theta = \{\theta_i | i = 1, 2, \dots, N_\Theta\}$  é um conjunto de  $N_\Theta$  comandos.

A definição de PS pode então ser enunciada:

**Definição 8** Um provedor de serviços (PS) é uma tripla  $(\mathcal{M}_{PS}(a), b(s, a), c(s, a))$  em que:

- $\mathcal{M}_{PS}(a)$  é uma cadeia de Markov estacionária controlável com conjunto de estados  $\mathcal{S} = \{s_i | i = 1, 2, \dots, S\}$ , conjunto de comandos  $\mathcal{A} = \{a_i, | i = 1, 2, \dots, A\}$  e matriz de probabilidades de transição  $\mathbf{P}^{PS}(a)$ ;

- $b(s, a)$  é uma função  $b : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ , que representa uma taxa de serviço associada a cada estado  $s \in \mathcal{S}$  e comando  $a \in \mathcal{A}$ . Tal taxa de serviço é a probabilidade de o PS completar o atendimento de uma requisição de serviço num período, dado que o PS está no estado  $s$  e comando  $a$  foi passado no início do período;
- $c(s, a)$  é uma função  $c : \mathcal{S} \times \mathcal{A} \rightarrow \mathbb{R}$ , que representa uma métrica de consumo de potência, corrente ou energia associada a cada estado  $s \in \mathcal{S}$  e comando  $a \in \mathcal{A}$ , i.e. o consumo (em unidades arbitrárias:  $W$ ,  $A$  ou  $J$ ) do PS num período, dado que o comando  $a$  foi tomado e o PS está no estado  $s$ .

Se um PS de dois estados  $\mathcal{S} = \{s_i, s_j\}$  está no estado  $s_i$ , e o comando  $a$  é mantido até que o PS transite para o estado  $s_j$ , o valor esperado do tempo de transição entre esses estados é calculado da mesma forma como foi realizado na Equação (2.13) e é apresentado na Equação (2.14).

$$\bar{t}_{s_i, s_j}(a) = T \sum_{k=1}^{\infty} k p_{s_i, s_j}^{PS}(a) \left(1 - p_{s_i, s_j}^{PS}(a)\right)^{k-1} = \frac{T}{p_{s_i, s_j}^{PS}(a)} \quad (2.14)$$

A carga de trabalho do sistema é modelada como um requisitante de serviços:

**Definição 9** Um requisitante de serviços (RS) é um par  $(\mathcal{M}_{RS}, z(r))$  em que:

- $\mathcal{M}_{RS}$  é uma cadeia de Markov estacionária com conjunto de estados  $\mathcal{R} = \{r_i | i = 0, 1, \dots, (R-1)\}$  e matriz de transição  $\mathbf{P}^{RS}$ ;
- $z(r)$  é uma função  $z : \mathcal{R} \rightarrow \mathbb{N}$  que representa o número de requisições lançadas por período pelo requisitante de serviços quando no estado  $r$ .

**Exemplo 1** Considere um RS com conjunto de estados  $\{r_0, r_1\}$ , e com função  $z(r)$  que representa uma correspondência um-para-um entre as imagens  $z$  e os estados do RS:  $z(r_0) = 0$ ,  $z(r_1) = 1$ . Então é usado 0 ao invés de  $r_0$ , e 1 ao invés de  $r_1$ . O modelo do RS é mostrado na Figura 2.2 e sua matriz de transição na Equação (2.15).

$$P^{RS} = \begin{matrix} & \begin{matrix} 0 & 1 \end{matrix} \\ \begin{matrix} 0 \\ 1 \end{matrix} & \begin{bmatrix} 0,85 & 0,15 \\ 0,80 & 0,20 \end{bmatrix} \end{matrix} \quad (2.15)$$

A FS armazena requisições advindas do RS que não puderam ser atendidas imediatamente pelo PS.

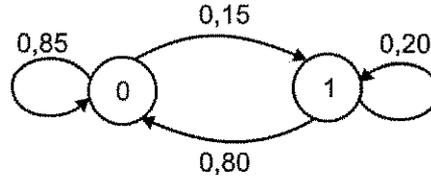


Figura 2.2: Exemplo de modelo em cadeia de Markov para um requisitante de serviços.

**Definição 10** Uma fila de serviço (FS) de tamanho  $Q - 1$  é uma cadeia de Markov estacionária controlável  $\mathcal{M}_{FS}(a, s, r)$  com conjunto de estados  $\mathcal{Q} = \{q_i | i = 0, 1, \dots, (Q - 1)\}$ , variável de controle  $\theta = (a, s, r)$ , conjunto de comandos  $\Theta = \mathcal{A} \times \mathcal{S} \times \mathcal{R}$  e matriz de transição  $\mathbf{P}^{FS}(a, s, r)$ . As probabilidades de transição são definidas na Equação (2.16).

$$p_{q_i, q_j}^{FS}(a, s, r) = \begin{cases} 1 - b(a, s), & \text{se } j = i + z(r) \text{ e } 0 < i + z(r) < Q \\ b(a, s), & \text{se } j = i + z(r) - 1 \text{ e } 0 \leq i + z(r) - 1 < Q \\ 1, & \text{se } j = i = 0 \text{ e } z(r) = 0 \\ 1, & \text{se } j = Q - 1 \text{ e } i + z(r) \geq Q \\ 0, & \text{caso contrário} \end{cases} \quad (2.16)$$

Os casos da expressão (2.16) podem ser melhor entendidos da seguinte forma:

- Caso  $j = i + z(r)$  e  $0 < i + z(r) < Q$ : a probabilidade de transição é  $1 - b(a, s)$  pois o PS não serviu nenhuma requisição ( $j = i + z(r)$ ), e o número de requisições que chegaram ( $z(r)$ ) mais as que já estavam na fila ( $i$ ) não excede o tamanho da fila ( $Q - 1$ ). Na Figura 2.3 é mostrado um exemplo onde um RS gera uma requisição que é armazenada numa FS de comprimento 2 que já possuía uma requisição pendente, pois o PS ainda não atendeu nenhuma delas;
- Caso  $j = i + z(r) - 1$  e  $0 \leq i + z(r) - 1 < Q$ : a probabilidade de transição é  $b(a, s)$  pois o PS serviu uma requisição ( $j = i + z(r) - 1$ ), e a soma do número de requisições que estavam da fila ( $i$ ) mais as que chegaram ( $z(r)$ ) menos a que foi servida não excede o tamanho da fila ( $Q - 1$ ). Na Figura 2.4 é mostrado um exemplo onde um RS gera uma requisição que é armazenada numa FS de comprimento 2 que já possuía uma requisição pendente a qual é atendida pelo PS, e a cadeia de Markov da FS permanece no mesmo estado;
- Caso  $j = i = 0$  e  $z(r) = 0$ : a probabilidade de transição é 1 pois este é o caso limite em que a fila estava vazia e assim permanece ( $j = i = 0$ ) e não aparece nenhuma nova requisição ( $z(r) = 0$ ). Na Figura 2.5 é mostrado um exemplo onde um RS não gera nenhuma requisição a ser armazenada numa FS de comprimento 2 que estava vazia e assim permanece;

- Caso  $j = Q - 1$  e  $i + z(r) \geq Q$ : a probabilidade de transição é 1 pois este é o caso limite em que a fila se torna cheia ( $j = Q - 1$ ) e a soma do número de requisições que estavam da fila ( $i$ ) mais as que chegaram ( $z(r)$ ) excede o tamanho da fila. Uma perda de requisição ocorre sempre que uma nova requisição do RS não pode ser colocada na fila porque esta já está cheia. Na Figura 2.6 é mostrado um exemplo com um RS que gera duas requisições e uma FS de comprimento 2 que já possuía uma requisição pendente; a fila fica cheia e a requisição que excedeu o comprimento da fila ou é atendida pelo PS ou é perdida;

**Exemplo 2** Considere uma FS de comprimento 1 e conjunto de estados  $\mathcal{Q} = \{q_0, q_1\}$ , i.e., em  $q_0$  a FS está vazia, em  $q_1$  a FS tem uma requisição de serviço pendente e está conseqüentemente cheia pois o comprimento máximo da fila é 1. Sua matriz de transição  $\mathbf{P}^{FS}(a, s, r)$  é mostrada na expressão (2.17) para  $r = 0$  e na expressão (2.18) para  $r = 1$ . Como pode ser observado:

- $p_{q_0, q_0}^{FS}(a, s, r = 0) = 1$  porque a fila está vazia e assim permanece ( $j = i = 0$ ) e não aparece nenhuma nova requisição;
- $p_{q_0, q_1}^{FS}(a, s, r = 0) = 0$  porque que a fila está vazia e não há nenhuma nova requisição;
- $p_{q_1, q_0}^{FS}(a, s, r = 1) = 0$  porque há duas requisições a serem atendidas e somente uma pode ser atendida;
- $p_{q_1, q_1}^{FS}(a, s, r = 1) = 1$  porque a fila se está cheia ( $j = Q - 1$ ) e a soma do número de requisições que estavam da fila (1) mais as que chegaram ( $r = 1$ ) excede o tamanho da fila.

$$p_{q_i, q_j}^{FS}(a, s, r = 0) = \begin{matrix} & q_0 & q_1 \\ \begin{matrix} q_0 \\ q_1 \end{matrix} & \begin{bmatrix} 1 & 0 \\ b(a, s) & 1 - b(a, s) \end{bmatrix} \end{matrix} \quad (2.17)$$

$$p_{q_i, q_j}^{FS}(a, s, r = 1) = \begin{matrix} & q_0 & q_1 \\ \begin{matrix} q_0 \\ q_1 \end{matrix} & \begin{bmatrix} b(a, s) & 1 - b(a, s) \\ 0 & 1 \end{bmatrix} \end{matrix} \quad (2.18)$$

**Exemplo 3** Considere uma FS de comprimento 2 e conjunto de estados  $\mathcal{Q} = \{q_0, q_1, q_2\}$  cuja cadeia de Markov é ilustrada na Figura 2.7. Sua matriz de transição  $\mathbf{P}^{FS}(a, s, r)$  é mostrada nas expressões (2.19), (2.20) e (2.21). Como pode ser observado:

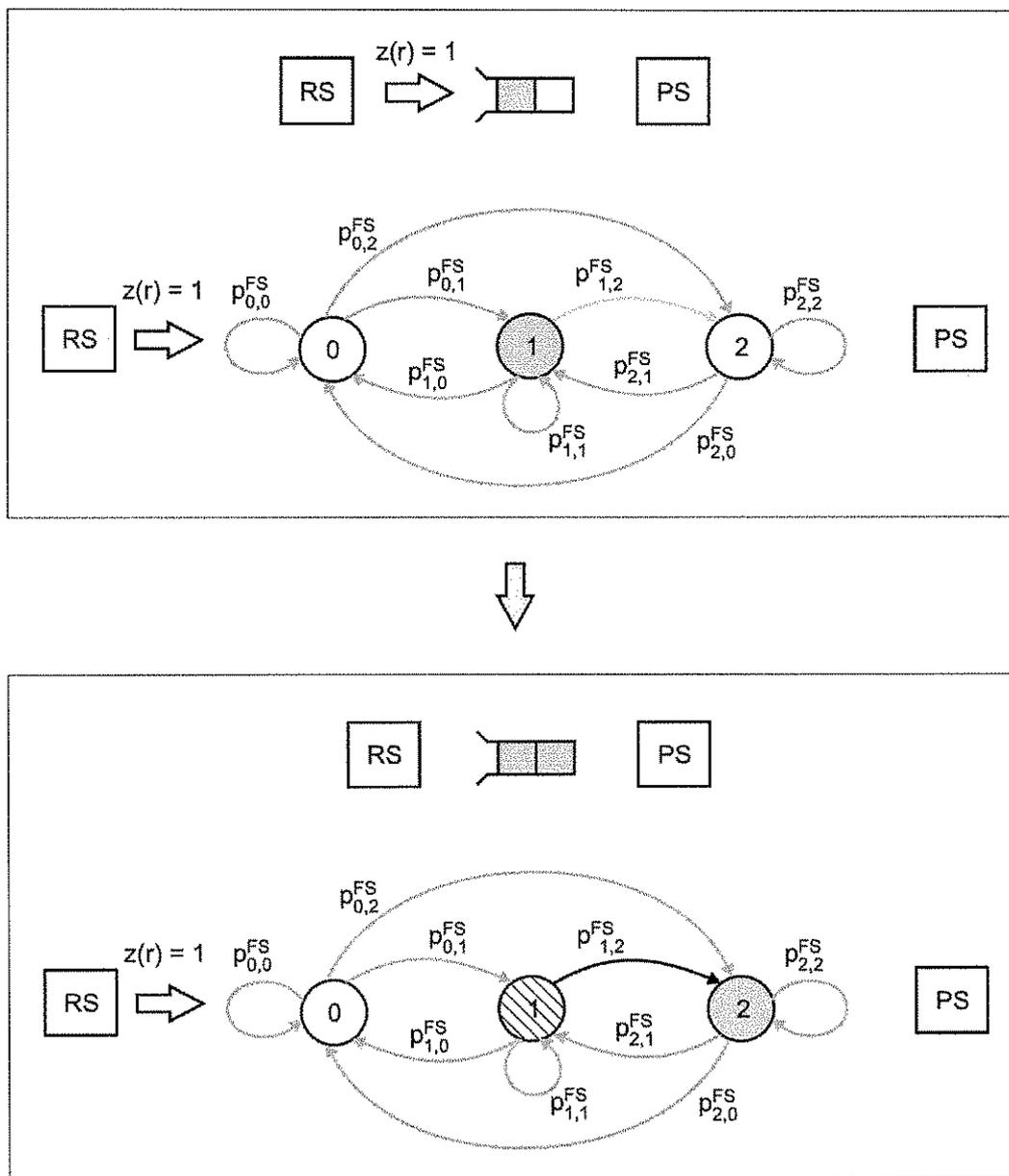


Figura 2.3: O PS não serviu nenhuma requisição e o número de requisições que chegaram mais as que já estavam na fila não excede o tamanho da fila.

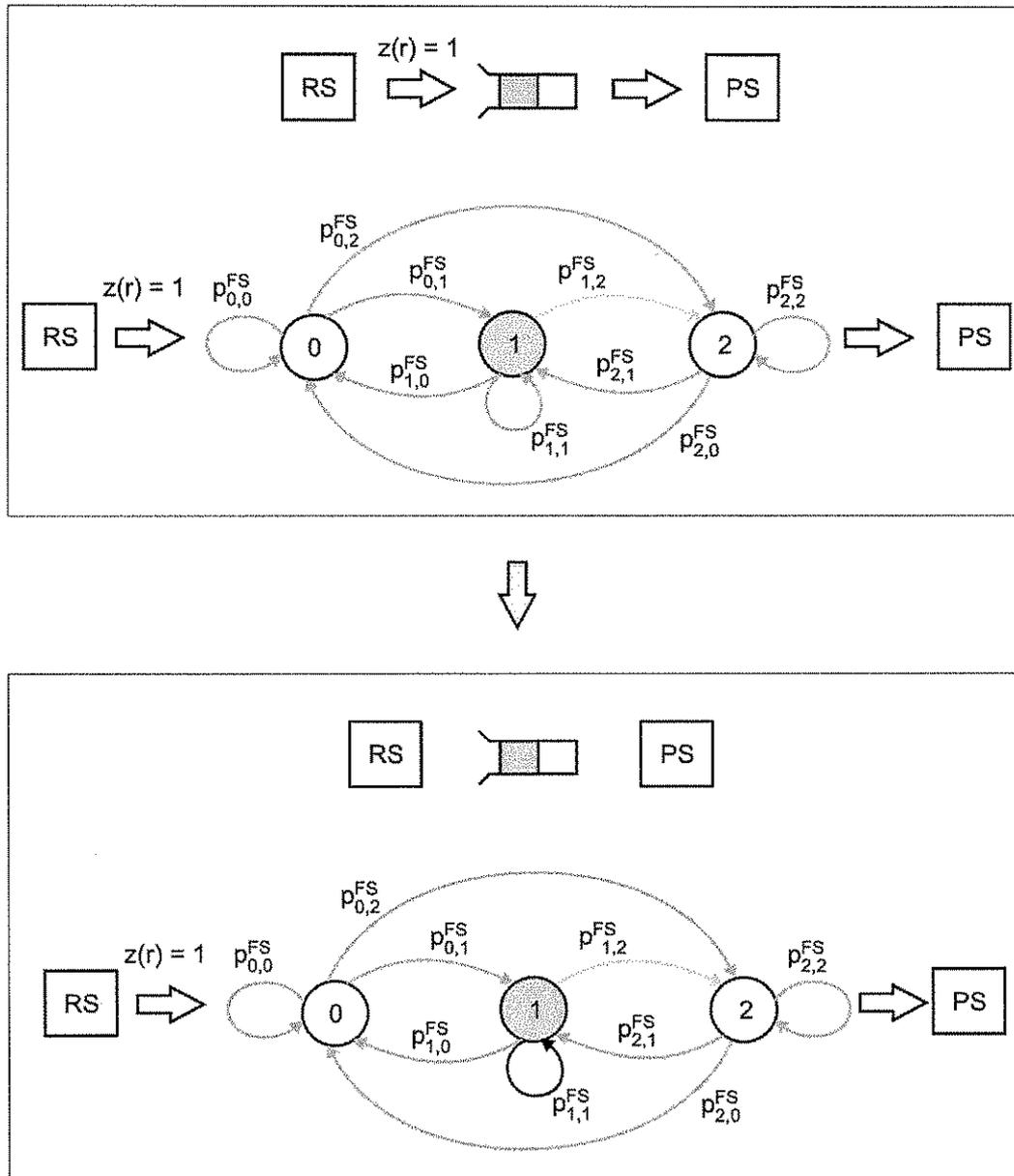


Figura 2.4: O PS serviu uma requisição e a soma do número de requisições que estavam na fila mais as que chegaram menos a que foi servida não excede o tamanho da fila.

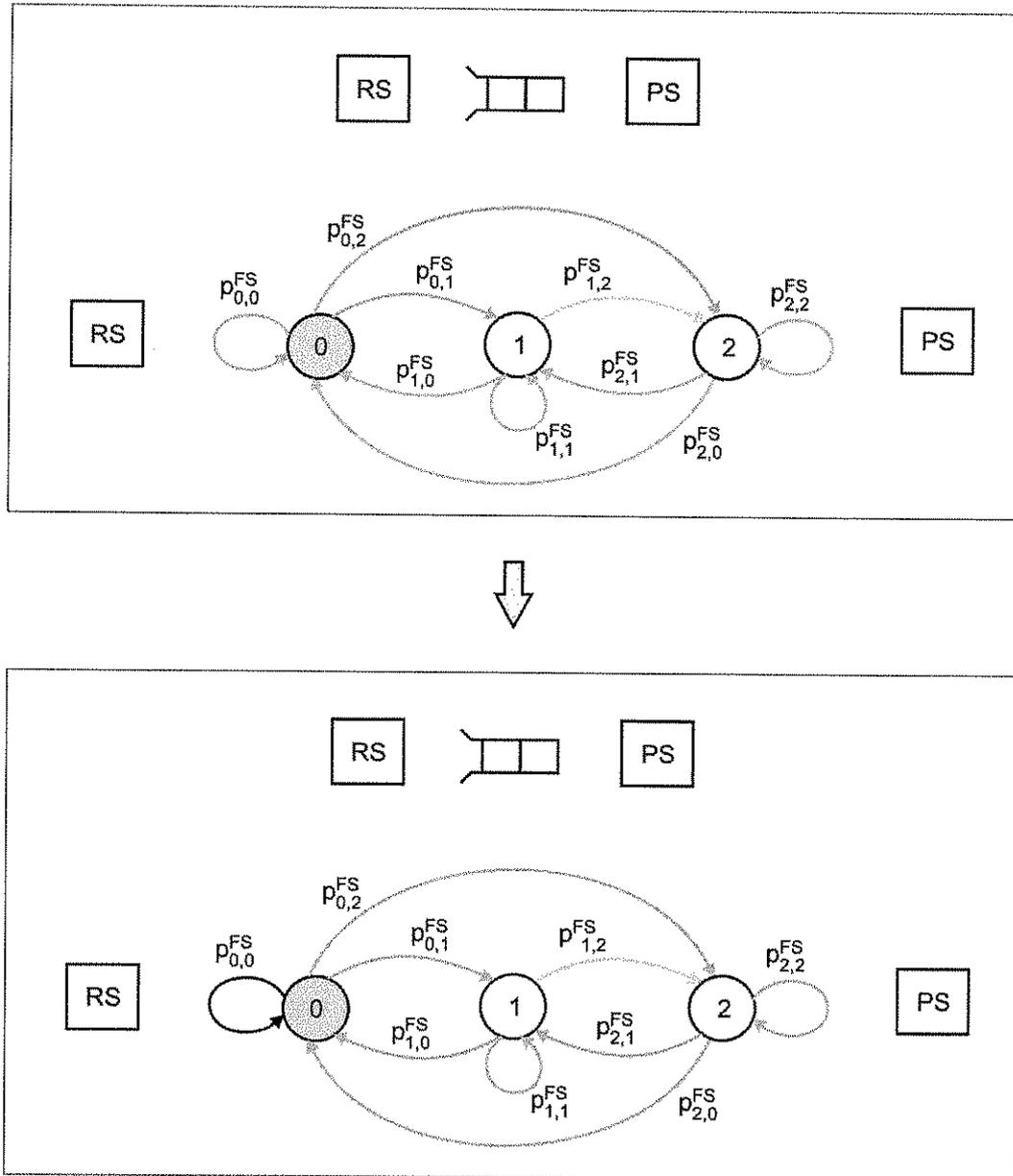


Figura 2.5: A fila estava vazia e assim permanece pois não aparece nenhuma nova requisição.

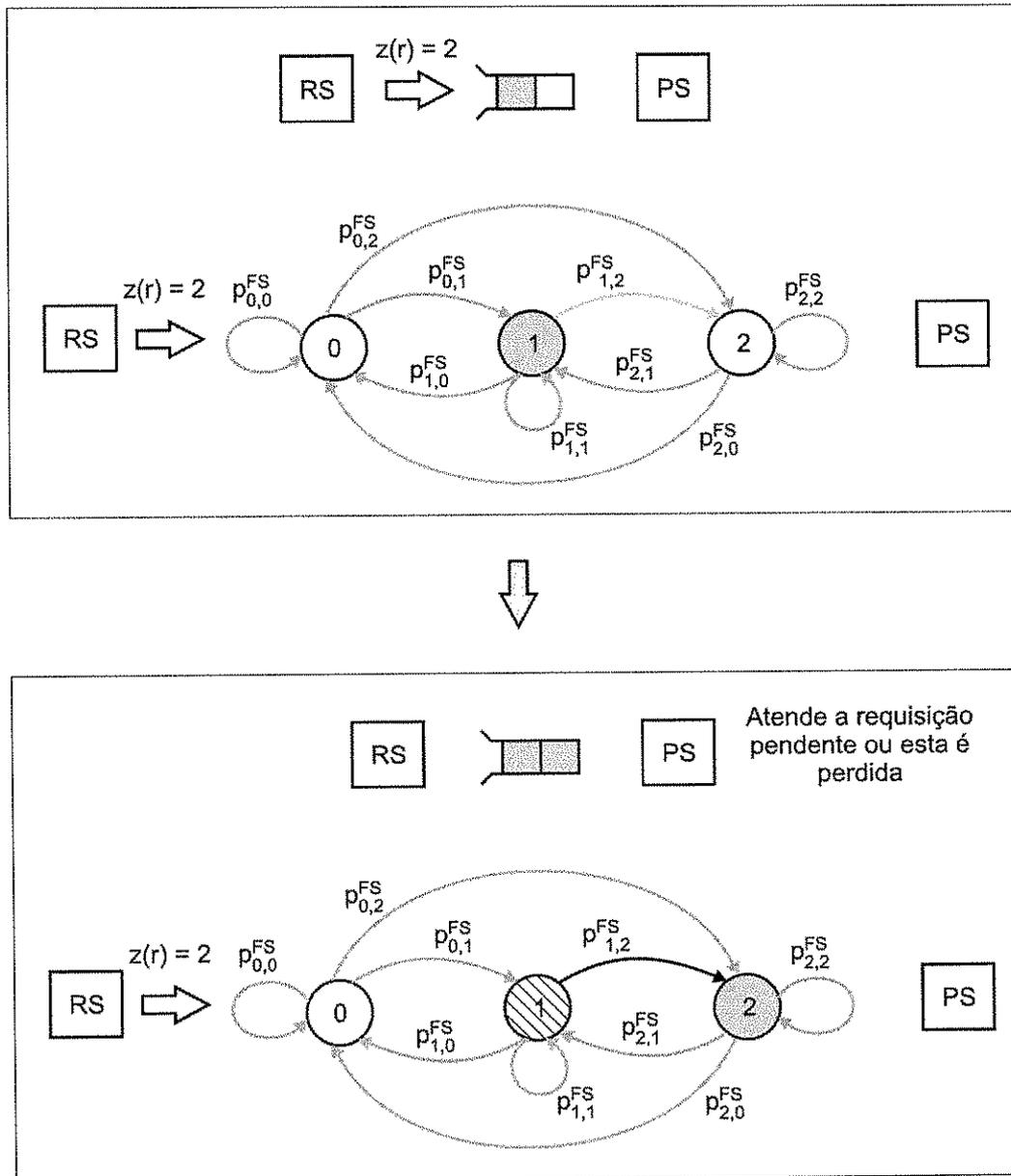


Figura 2.6: A fila se torna cheia e a soma do número de requisições que estavam na fila mais as que chegaram excede o tamanho da fila.

- $p_{q_0, q_0}^{FS}(a, s, r = 0) = 1$  porque a fila está vazia e assim permanece ( $j = i = 0$ ) pois não há nenhuma nova requisição;
- $p_{q_2, q_2}^{FS}(a, s, r = 1) = 1$  porque a fila está cheia ( $j = Q - 1$ ) e a soma do número de requisições que estavam da fila (2) mais as que chegaram ( $r = 1$ ) excede o tamanho da fila.
- $p_{q_1, q_2}^{FS}(a, s, r = 2) = 1$  porque a fila se torna cheia ( $j = Q - 1$ ) e a soma do número de requisições que estavam da fila (1) mais as que chegaram ( $r = 2$ ) excede o tamanho da fila.
- $p_{q_2, q_2}^{FS}(a, s, r = 2) = 1$  porque a fila está cheia ( $j = Q - 1$ ) e a soma do número de requisições que estavam da fila (2) mais as que chegaram ( $r = 2$ ) excede o tamanho da fila.

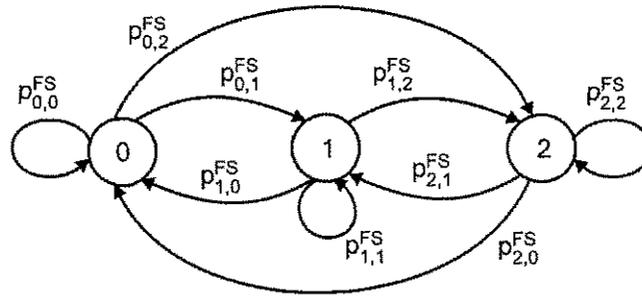


Figura 2.7: FS de comprimento 2.

$$p_{q_i, q_j}^{FS}(a, s, r = 0) = \begin{matrix} & q_0 & q_1 & q_2 \\ \begin{matrix} q_0 \\ q_1 \\ q_2 \end{matrix} & \begin{bmatrix} 1 & 0 & 0 \\ b(a, s) & 1 - b(a, s) & 0 \\ 0 & b(a, s) & 1 - b(a, s) \end{bmatrix} \end{matrix} \quad (2.19)$$

$$p_{q_i, q_j}^{FS}(a, s, r = 1) = \begin{matrix} & q_0 & q_1 & q_2 \\ \begin{matrix} q_0 \\ q_1 \\ q_2 \end{matrix} & \begin{bmatrix} b(a, s) & 1 - b(a, s) & 0 \\ 0 & b(a, s) & 1 - b(a, s) \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \quad (2.20)$$

$$p_{q_i, q_j}^{FS}(a, s, r = 2) = \begin{matrix} & q_0 & q_1 & q_2 \\ \begin{matrix} q_0 \\ q_1 \\ q_2 \end{matrix} & \begin{bmatrix} 0 & b(a, s) & 1 - b(a, s) \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix} \end{matrix} \quad (2.21)$$

**Exemplo 4** Considere agora o caso geral de uma FS de comprimento  $Q$  e conjunto de estados  $\mathcal{Q} = \{q_0, q_1, q_2 \dots q_{Q-1}\}$ . Sua matriz de transição  $\mathbf{P}^{FS}(a, s, r)$  é mostrada nas expressões (2.22) a (2.26).

$$p_{q_i, q_j}^{FS}(a, s, r = 0) = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 & 0 \\ b(a, s) & 1 - b(a, s) & 0 & \dots & 0 & 0 \\ 0 & b(a, s) & 1 - b(a, s) & \dots & 0 & 0 \\ \vdots & & & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & b(a, s) & 1 - b(a, s) \end{bmatrix} \quad (2.22)$$

$$p_{q_i, q_j}^{FS}(a, s, r = 1) = \begin{bmatrix} b(a, s) & 1 - b(a, s) & 0 & 0 & \dots & 0 & 0 \\ 0 & b(a, s) & 1 - b(a, s) & 0 & \dots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \dots & b(a, s) & 1 - b(a, s) \\ 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (2.23)$$

$$p_{q_i, q_j}^{FS}(a, s, r = 2) = \begin{bmatrix} 0 & b(a, s) & 1 - b(a, s) & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & b(a, s) & 1 - b(a, s) & 0 & \dots & 0 & 0 \\ \vdots & & & & & & & \\ 0 & 0 & 0 & 0 & 0 & \dots & b(a, s) & 1 - b(a, s) \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (2.24)$$

$$p_{q_i, q_j}^{FS}(a, s, r = Q - 1) = \begin{bmatrix} 0 & 0 & \dots & b(a, s) & 1 - b(a, s) \\ 0 & 0 & \dots & 0 & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (2.25)$$

$$p_{q_i, q_j}^{FS}(a, s, r \geq Q) = \begin{bmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 1 \\ \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \quad (2.26)$$

**Definição 11** Seja  $\mathcal{H}_n = (\mathcal{S} \times \mathcal{R} \times \mathcal{Q})^n$  o conjunto de todas as possíveis realizações do sistema do instante  $t_1$  a  $t_n$ . Assim um elemento  $H_n \in \mathcal{H}_n$  representa a história do sistema do instante  $t_1$  a  $t_n$ , ou seja,  $H_n = ((s_1, r_1, q_1), (s_2, r_2, q_2), \dots, (s_n, r_n, q_n))$ . Um gerenciador de energia (GE) observa a história  $H_n$  do sistema até o instante  $t_n$  e com base nela passa um comando  $a \in \mathcal{A}$  ao PS a cada instante de tempo  $t_n$ .

**Exemplo 5** Considere um PS com conjunto de estados {ligado, desligado}, e um GE com comandos {ligar, desligar}, respectivamente. Cada probabilidade de o PS transitar de um estado de energia para outro é uma função do comando que o GE passa, como mostrado na Figura 2.8. A matriz  $c(s, a)$  da Equação (2.27) especifica o consumo de corrente (em mA) do PS no estado  $s \in \{\text{ligado}, \text{desligado}\}$ , e quando o comando  $a \in \{\text{ligar}, \text{desligar}\}$  é passado pelo GE. A matriz  $b(s, a)$  da Equação (2.28) especifica a taxa de serviço do PS no estado  $s \in \{\text{ligado}, \text{desligado}\}$  e quando o comando  $a \in \{\text{ligar}, \text{desligar}\}$  é passado pelo GE, i.e., a probabilidade de servir uma requisição num período.

$$c(s, a) = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} \text{ligado} \\ \text{desligado} \end{matrix} & \begin{bmatrix} 200 & 250 \\ 250 & 0 \end{bmatrix} \end{matrix} \quad (2.27)$$

$$b(s, a) = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} \text{ligado} \\ \text{desligado} \end{matrix} & \begin{bmatrix} 0,8 & 0 \\ 0 & 0 \end{bmatrix} \end{matrix} \quad (2.28)$$

$$P^{PS}(\text{ligar}) = \begin{matrix} & \begin{matrix} \text{ligado} & \text{desligado} \end{matrix} \\ \begin{matrix} \text{ligado} \\ \text{desligado} \end{matrix} & \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \end{matrix} \quad (2.29)$$

$$P^{PS}(\text{desligar}) = \begin{matrix} & \begin{matrix} \text{ligado} & \text{desligado} \end{matrix} \\ \begin{matrix} \text{ligado} \\ \text{desligado} \end{matrix} & \begin{bmatrix} 0,8 & 0,2 \\ 0 & 1 \end{bmatrix} \end{matrix} \quad (2.30)$$

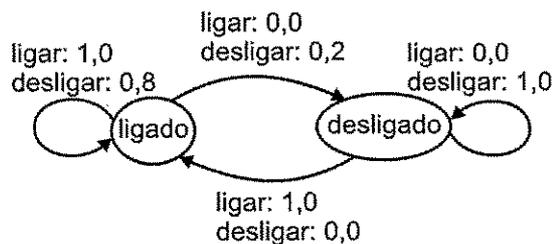


Figura 2.8: Exemplo de modelo em cadeia de Markov para um provedor de serviços.

O *modelo do sistema* como um todo é então a composição das cadeias de Markov do provedor de serviço, requisitante de serviços e fila de serviços, resultando numa cadeia de Markov controlável com conjunto de estados  $\mathcal{X} = \mathcal{S} \times \mathcal{R} \times \mathcal{Q}$ . Um elemento  $p_{x_t, x_j}(a)$  de sua matriz de transição  $\mathbf{P}(a)$  é calculado segundo a Equação (2.31).

$$p_{x_i, x_j}(a) = \text{Prob}(x_j = (s', r', q') | x_i = (s, r, q)) = p_{s, s'}^{PS}(a) \cdot p_{r, r'}^{RS} \cdot p_{q, q'}^{FS}(a, s, r) \quad (2.31)$$

Uma *decisão*  $\delta(H_n) = \{p_a(H_n) | a \in \mathcal{A}\}$  no instante de tempo  $t_n$  é um conjunto de funções  $p_a : \mathcal{H}_n \rightarrow [0, 1]$  com a propriedade  $\sum_{p_a \in \delta} p_a(H_n) = 1$ . No presente contexto, uma decisão  $\delta(H_n)$  é uma distribuição de probabilidade discreta que a cada comando  $a \in \mathcal{A}$  atribui uma probabilidade  $p_a(H_n)$ . Utiliza-se a notação  $\delta^{(n)} = \delta(H_n)$ . Na prática, no início de um período  $n$ , com base na história  $H_n$  do sistema, o gerenciador de energia passa ao PS um comando  $a$  com probabilidade  $p_a(H_n)$ .

A *matriz de transição do sistema* dada a decisão  $\delta^{(n)}$  é calculada segundo a Equação (2.32).

$$\mathbf{P}_{\delta^{(n)}} = \sum_{p_a \in \delta^{(n)}} p_a \mathbf{P}(a) \quad (2.32)$$

Uma *política de gerenciamento de energia* é definida como a seqüência discreta  $\pi = [\delta^{(1)}, \delta^{(2)}, \dots]$  de decisões tomadas nos instantes de tempo  $[1, 2, \dots]$ . Uma *política estacionária* é aquela em que é tomada a mesma decisão  $\delta^{(n)} = \delta$  em todos os instantes de tempo  $t_i$ ,  $i = 1, 2, \dots$  resultando então numa política  $\pi = [\delta, \delta, \dots]$ . Assim *políticas estacionárias de Markov* são aquelas em que a decisão não depende da história  $H_n$  mas somente do estado do sistema  $x = (s, r, q)$  no instante  $t_n$ .

Dado que o sistema está num estado  $x = (s, r, q)$  e é tomada uma decisão  $\delta_x$  podem ser definidas *métricas de custo*. Sendo  $c(s, a)$  a potência consumida pelo PS quando no estado  $s$  e o comando  $a$  é escolhido, tem-se a métrica de *nível esperado de consumo de potência*  $\bar{c}(x, \delta_x) = \sum_{p_a \in \delta_x} p_a c(s, a)$ . A métrica de *penalidade de desempenho* por unidade de tempo  $d(x)$  está associada ao tempo de espera e o número de trabalhos na fila. É possível definir por simplicidade  $d(x) = q$ , o número de requisições na fila. Vetores de consumo e penalidade de desempenho são montados de acordo com a Equação (2.33).

$$\bar{c} := \begin{pmatrix} \bar{c}(x_1, \delta_{x_1}) \\ \vdots \\ \bar{c}(x_X, \delta_{x_X}) \end{pmatrix} \quad d_\delta := \begin{pmatrix} d(x_1) \\ \vdots \\ d(x_X) \end{pmatrix} \quad (2.33)$$

Seja um sistema em que  $\mathbf{p}^{(1)}$  é um vetor linha de dimensão  $X$  que representa a distribuição de probabilidade dos estados no instante inicial  $t_1$ . A matriz de transição do

sistema é  $\mathbf{P}(a)$  e ele possui a política  $\pi$ . Assim a distribuição de probabilidades do sistema no instante  $t_n$  é calculada como  $\mathbf{p}^{(n)} = \mathbf{p}^{(1)}\mathbf{P}_\pi^{(n-1)}$  e, em conseqüência, os valores esperados da penalidade de desempenho e do consumo no tempo  $n$  são obtidos segundo as Equações (2.34) e (2.35) respectivamente.

$$E[\mathbf{d}_{\delta^{(n)}}^\pi] = \mathbf{p}^{(1)}\mathbf{P}_\pi^{(n-1)}\mathbf{d}_{\delta^{(n)}} \quad (2.34)$$

$$E[\bar{\mathbf{c}}_{\delta^{(n)}}^\pi] = \mathbf{p}^{(1)}\mathbf{P}_\pi^{(n-1)}\bar{\mathbf{c}}_{\delta^{(n)}} \quad (2.35)$$

Assume-se que o sistema será utilizado durante um horizonte de tempo finito e aleatório  $\hat{N}$ , que é denominado *tempo de parada*. Num sistema embarcado alimentado por bateria, por exemplo, o tempo de parada é o intervalo de tempo desde o início da utilização do sistema, quando a bateria está com carga total, até o instante em que a bateria é considerada descarregada e o sistema não mais funciona. A janela de interesse para o problema de otimização é o período  $[0, \hat{N}]$ , sendo  $\hat{N}$  aleatório devido à incerteza sobre a duração exata desse período. Assume-se que  $\hat{N}$  é uma variável aleatória geometricamente distribuída com parâmetro  $1 - \xi$ , onde  $0 < \xi < 1$ , e valor esperado igual a  $E[\hat{N}] = (1 - \xi)^{-1}$ . O parâmetro  $1 - \xi$  representa a probabilidade a cada período de o tempo de parada ser alcançado. Então a probabilidade de o sistema continuar funcionando a cada período é  $\xi$ , denominado *fator de desconto*, pois “desconta” o efeito do futuro introduzindo no modelo um horizonte de tempo finito. As expressões (2.34) e (2.35) são reescritas como (2.36) e (2.37) respectivamente.

$$E[\mathbf{d}_{\delta^{(n)}}^\pi] = \xi^{n-1}\mathbf{p}^{(1)}\mathbf{P}_\pi^{(n-1)}\mathbf{d}_{\delta^{(n)}} \quad (2.36)$$

$$E[\bar{\mathbf{c}}_{\delta^{(n)}}^\pi] = \xi^{n-1}\mathbf{p}^{(1)}\mathbf{P}_\pi^{(n-1)}\bar{\mathbf{c}}_{\delta^{(n)}} \quad (2.37)$$

O *problema de otimização* (PO) consiste na escolha da política que minimiza o consumo sob restrições de desempenho (ou maximiza o desempenho sob restrição de consumo) (4). O *problema de otimização de desempenho sob restrição de consumo PO1* é formulado em (2.38), e o *problema de otimização de consumo sob restrição de desempenho PO2* em (2.39). Cada um desses é formulado como um problema de otimização de programação linear (PL), sendo possível resolvê-lo de forma exata e em tempo polinomial (em  $A \cdot X$ ). Como descrito por Paleologo et al.(4), o problema PO2 pode ser resolvido pelo programa linear representado em (2.40), no qual as incógnitas  $f_{x,a}$  são denominadas *freqüências estado-ação* e representam o número esperado de vezes que o sistema está no estado  $x$  e o comando  $a$  é passado pelo GE.

$$\text{PO1} : \min_{\pi} \sum_{n=1}^{\infty} E[\mathbf{d}_{\delta^{(n)}}^\pi] \quad \text{tal que} \quad \sum_{n=1}^{\infty} E[\bar{\mathbf{c}}_{\delta^{(n)}}^\pi] \leq C \quad (2.38)$$

$$\text{PO2 : } \min_{\pi} \sum_{n=1}^{\infty} E [\bar{c}_{\delta(n)}^{\pi}] \text{ tal que } \sum_{n=1}^{\infty} E [d_{\delta(n)}^{\pi}] \leq D \quad (2.39)$$

$$\begin{aligned} \text{PL1 : } \quad & \min \sum_{x \in \mathcal{X}} \sum_{a \in \mathcal{A}} f_{x,a} c(x, a) & (2.40) \\ \text{tal que } & \sum_{a \in \mathcal{A}} f_{x,a} - \xi \sum_{y \in \mathcal{X}} \sum_{a \in \mathcal{A}} p_{y,x}(a) f_{y,a} = p_x^{(1)}, \forall x \in \mathcal{X} \\ & \sum_{x \in \mathcal{X}} \sum_{a \in \mathcal{A}} f_{x,a} d(x) \leq D \\ & f_{x,a} \geq 0, \forall x \in \mathcal{X}, a \in \mathcal{A} \end{aligned}$$

Para resolver o programa linear PL1, é possível utilizar: a ferramenta PCx, baseada num algoritmo de ponto interior (18); ou a ferramenta *linprog* do *Toolbox* de Otimização do Matlab (19), cuja implementação para otimização em larga escala é baseada também num algoritmo de ponto interior. A probabilidade  $m_{x,a}$  de o GE passar o comando  $a$  quando o sistema está no estado  $x$  é estimada em (2.41). As probabilidades  $m_{x,a}$  são organizadas numa tabela de decisão  $M_{\pi}$  (Equação (2.42)) em que as linhas variam com os estados  $x_1, x_2 \dots x_X$  do sistema e as colunas com os comandos  $a_1, a_2, \dots a_A$ . Em tempo de execução, a política de energia toma a linha de  $M_{\pi}$  correspondente ao estado atual  $x$  do sistema:  $(m_{x,a_1} \ m_{x,a_2} \ \dots \ m_{x,a_A})$ , denominada *decisão*. Essa linha representa a distribuição de probabilidades de o GE passar o comando  $a$  com probabilidade  $m_{x,a}$  dado que o estado do sistema é  $x$ . Deseja-se escolher o comando  $a$  que o GE vai passar ao PS. Para isso é realizado o sorteio de um número aleatório  $D$  uniformemente distribuído no intervalo  $[0, 1]$  e escolhido o comando  $a_i$  tal que  $\sum_{k=1}^{i-1} m_{x,a_k} < D \leq \sum_{k=1}^i m_{x,a_k}$  caso  $0 < D \leq 1$ , ou  $a_1$  caso  $D = 0$  (veja a Figura 2.9).

$$m_{x,a} = \frac{f_{x,a}}{\sum_{a' \in \mathcal{A}} f_{x,a'}} \quad (2.41)$$

$$M_{\pi} = \begin{bmatrix} m_{x_1,a_1} & m_{x_1,a_2} & \dots & m_{x_1,a_A} \\ m_{x_2,a_1} & m_{x_2,a_2} & \dots & m_{x_2,a_A} \\ \vdots & \vdots & \vdots & \vdots \\ m_{x_X,a_1} & m_{x_X,a_2} & \dots & m_{x_X,a_A} \end{bmatrix} \quad (2.42)$$

**Exemplo 6** Considere o RS do exemplo 1, a FS do exemplo 2, e o PS e GE do exemplo 5. Deseja-se simular esse sistema durante 1000 passos de simulação, o que resulta num fator de desconto  $\xi = 1 - 1/E[\hat{N}] = 1 - 1/1000 = 0,999$ . Deseja-se também que o número médio de requisições pendentes na FS seja de no máximo 50%, e que a perda de requisições média seja de no máximo 10%.

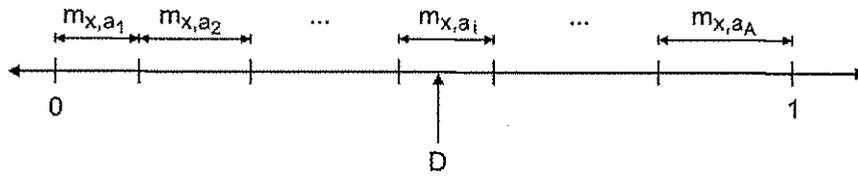


Figura 2.9: Escolha do comando que o GE passará ao PS.

A resolução do programa linear representado em (2.40) através da ferramenta linprog oferece a tabela de decisão  $M_\pi$  apresentada em (2.43). Quando o PS está ligado e não surgem novas requisições nem há requisições pendentes na fila i.e. estado (ligado,  $r_0, q_0$ ), o sistema está inativo e o GE manda o PS ser desligado com probabilidade 1 para fins de economizar energia. Nos demais estados em que o PS está ligado mas o sistema está ativo, (ligado,  $r_0, q_1$ ), (ligado,  $r_1, q_0$ ), (ligado,  $r_1, q_1$ ), o GE manda o PS continuar ligado com probabilidade 1 para que as requisições pendentes sejam atendidas. Quando o PS está desligado e não surgem novas requisições nem há requisições pendentes na fila i.e. estado (desligado,  $r_0, q_0$ ), o sistema está inativo e o GE manda o PS continuar desligado com probabilidade 1 para fins de continuar economizando energia. No estado (desligado,  $r_0, q_1$ ), o GE manda o PS ligar com probabilidade 0,1086 para que o PS possa atender a requisições pendentes quando ligar. Para os estados (desligado,  $r_1, q_0$ ) e (desligado,  $r_1, q_1$ ), o GE manda o PS continuar desligado com probabilidade 1, algo não desejável, dado que há requisições pendentes, mas contribuirá para a redução do consumo de energia sob o aumento de penalidade de desempenho.

$$M_\pi = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} (\text{ligado}, r_0, q_0) \\ (\text{ligado}, r_0, q_1) \\ (\text{ligado}, r_1, q_0) \\ (\text{ligado}, r_1, q_1) \\ (\text{desligado}, r_0, q_0) \\ (\text{desligado}, r_0, q_1) \\ (\text{desligado}, r_1, q_0) \\ (\text{desligado}, r_1, q_1) \end{matrix} & \begin{pmatrix} 0,0000 & 1,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 0,0000 & 1,0000 \\ 0,1086 & 0,8914 \\ 0,0000 & 1,0000 \\ 0,0000 & 1,0000 \end{pmatrix} \end{matrix} \quad (2.43)$$

Na Figura 2.10 é mostrado o início da simulação do sistema. No passo 0 da simulação não há novas requisições e a fila está vazia, então no passo 1 o GE manda o PS desligar. Mas no passo 1 surge uma nova requisição e o GE no passo 2 manda o PS continuar ligado para servir essa requisição, que é enfileirada na FS no passo 2 e atendida pelo PS que está ligado. No passo 4 o GE manda o PS desligar, mas no passo 7 surge uma nova requisição que é enfileirada no passo 8 e prontamente atendida pelo PS. No passo 10 o GE

manda novamente o PS desligar. Como não surgem novas requisições, o PS finalmente desliga no passo 13. No passo 20 surge uma nova requisição que é enfileirada no passo 21, mas não pode ser atendida pois o PS está desligado. No passo 23 o GE manda o PS ligar, o que acontece no passo 24. Durante esses 25 passos de simulação, não ocorreu nenhuma perda de requisições, o que aconteceria se surgisse uma nova requisição e a fila já estivesse cheia.

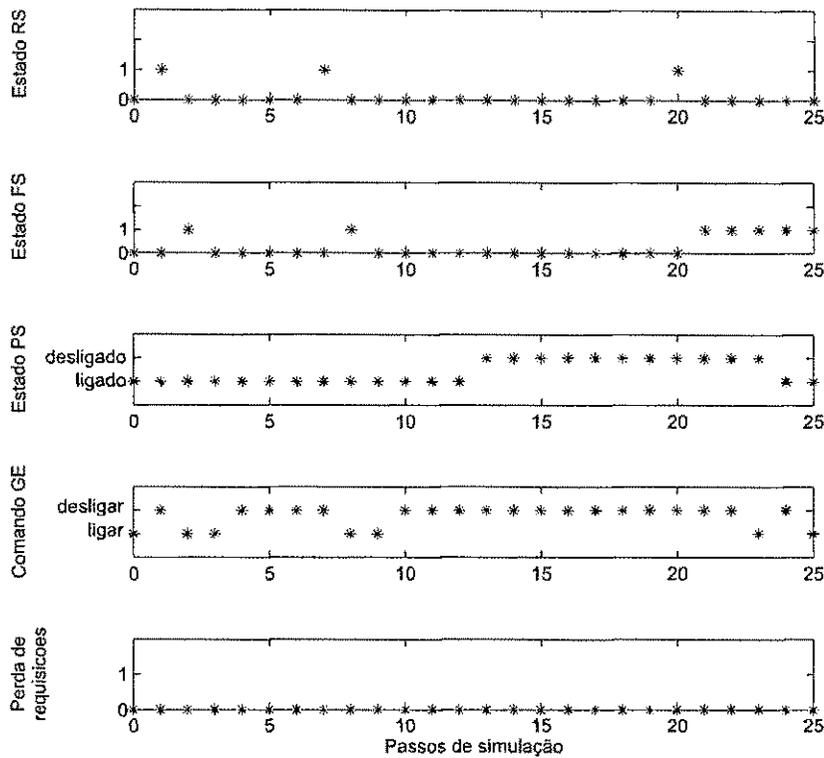


Figura 2.10: Início de uma simulação do sistema do exemplo 6

Ao final de 1000 passos de simulação, o consumo médio do sistema é de 110,55mA, o que representa uma redução de 44,72% em corrente média em relação a se o PS estivesse sempre ligado. O número médio de requisições pendentes na FS foi de 49,6%, menor que a utilizada como entrada para o problema de otimização (50%). A perda de requisições média foi de 7,5%, menor que a utilizada como entrada para o problema de otimização (10%).

É importante perceber que a abordagem de gerenciamento de energia através de processos de Markov no tempo discreto assegura a otimização do consumo sob as penalidades de desempenho na média, o que não impede que numa dada simulação ou execução prática o consumo se afaste do ótimo ou as penalidade de desempenho não sejam respeitadas. Assim, dada a natureza probabilística da fonte de requisições e da solução de gerenciamento de energia, só é possível esperar que pontos ótimos de consumo e das penalidades

de desempenho sejam atingidos na média.

O modelo de sistema de GDE em ambiente estacionário conhecido (4) assume que o modelo do RS é constante ao longo do tempo e é conhecido *a priori*, através de análises. Essas duas hipóteses impõem limitações ao GDE, pois cargas de trabalho reais podem apresentar não-estacionaridade. Para superar essas limitações, o modelo apresentado por Paleologo et al.(4) é estendido por Chung et al.(1) para o ambiente não-estacionário, que é descrito na Seção 2.2.

## 2.2 GDE em ambiente não-estacionário

O modelo do sistema com gerenciamento de energia utilizado por Chung et al.(1) é uma extensão do proposto por Paleologo et al.(4) para o caso não-estacionário. Admite-se que o RS tem suas probabilidades de transição de estados dependentes do tempo, tal como ilustrado na Figura 2.11 para um RS de dois estados.

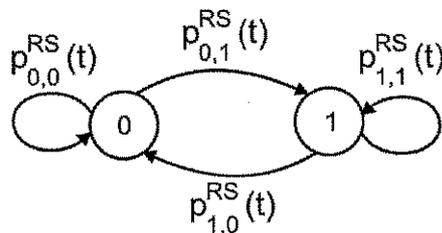


Figura 2.11: Exemplo de um RS não-estacionário.

Denota-se uma carga de trabalho não-estacionária por  $U^l$  e ela é modelada como uma série de  $N_u$  cargas de trabalho estacionárias  $u_s$ ,  $s = 0, 1, \dots, N_u - 1$  que possuem diferentes probabilidades de requisição de usuário. Uma carga de trabalho não-estacionária é então representada por  $U^{N_u} = (u_0, u_1, \dots, u_{N_u-1})$ . Sendo  $R_i$  a probabilidade de um RS permanecer no estado  $i$ , i.e.  $R_i = p_{r_i, r_i}^{RS}$ , percebe-se que  $R_i$  para o caso não-estacionário é uma função da seqüência de cargas estacionárias dada. As mudanças nas probabilidades de transição do modelo de Markov não-estacionário serão acompanhadas através apenas da observação de uma janela temporal de tamanho finito e de transições no passado. São propostas duas abordagens baseadas em janelas para estimar as probabilidades de transição da carga de trabalho: uma com janela única e outra com multi-janela.

### 2.2.1 Abordagem com janela única

A predição das requisições de usuário é baseada em sua história recente, armazenada numa janela deslizante. Essa janela deslizante é denotada por  $W$  e composta por  $l_w$  posições  $W(i)$ ,  $i = 0, 1, \dots, l_w - 1$ , cada uma armazenando uma requisição anterior do usuário, i.e.,

$s_r \in 0, 1, \dots, S_r - 1$ . A cada intervalo de tempo, o conteúdo da janela é deslocado, i.e.,  $W(i+1) \leftarrow W(i)$ ,  $i = 0, 1, \dots, l_w - 2$  e  $W(0)$  armazena uma nova requisição de usuário do RS. Essa operação é ilustrada na Figura 2.12.

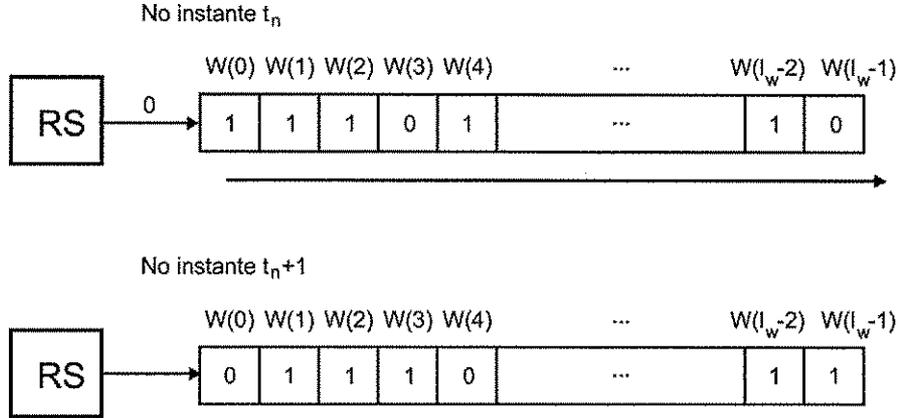


Figura 2.12: Exemplo de uso de uma janela única para RS de dois estados.

Seja “=” a operação de equivalência, i.e., o resultado é “1” quando os argumentos são iguais, e “0” caso contrário; “^” a operação de conjunção; e  $A_i = \sum_{k=1}^{l_w-1} (W(k) = i)$ . Então a estimativa da probabilidade de transição  $p_{r_i, r_j}^{RS}$ , denominada  $\hat{p}_{r_i, r_j}^{RS}$ , pode ser definido como na Equação (2.44).

$$\hat{p}_{r_i, r_j}^{RS}(t) = \begin{cases} \frac{1}{A_i} \sum_{k=1}^{l_w-1} [(W(k) = i) \wedge (W(k-1) = j)] & \text{se } A_i \neq 0 \\ 0 & \text{se } A_i = 0 \text{ e } i = j \\ \frac{1}{S_r-1} & \text{caso contrário.} \end{cases} \quad (2.44)$$

**Exemplo 7** Seja o RS com matriz de transição explicitada na equação (2.45). Foi utilizado o método de identificação do RS com a abordagem de janela única através de três comprimentos de janela: 10, 50 e 100, cujos resultados são mostrados nas Figuras 2.13, 2.14 e 2.15 respectivamente. Foi utilizado período de 1s e o tempo de simulação de 1000s. Em cada figura a linha tracejada representa a probabilidade real e a linha sólida é a identificação da probabilidade. Como esperado, quanto maior o comprimento da janela, melhores os resultados, mas as desvantagens são maior uso de memória (diretamente proporcional ao comprimento da janela) e tempo gasto nos cálculos para identificação.

$$P^{RS} = \begin{matrix} & 0 & 1 \\ \begin{matrix} 0 \\ 1 \end{matrix} & \begin{bmatrix} 0,1 & 0,9 \\ 0,6 & 0,4 \end{bmatrix} \end{matrix} \quad (2.45)$$

Seja  $e_{ij}(t) = \hat{p}_{r_i, r_j}^{RS}(t) - p_{r_i, r_j}^{RS}(t)$  o erro de estimação da probabilidade  $p_{r_i, r_j}^{RS}(t)$ . Dado que um RS é identificado ao longo de uma simulação ou execução durante os passos de  $k = 1 \dots n$ , o erro médio quadrático (EMQ) de estimação é calculado conforme (2.46).

$$EMQ_{ij} = \frac{1}{n} \sum_{k=1}^n (e_{ij}(t_k))^2 = \frac{1}{n} \sum_{k=1}^n \left[ \hat{p}_{r_i, r_j}^{RS}(t_k) - p_{r_i, r_j}^{RS}(t_k) \right]^2 \quad (2.46)$$

Uma comparação para o desempenho da estimação do RS através dos diferentes tamanhos de janela pode ser realizado através do cálculo do EMQ para as probabilidades  $p_{0,0}^{RS}(t)$  e  $p_{1,1}^{RS}(t)$ , conforme apresentado na Tabela 2.1. Percebe-se que, quanto maior o comprimento da janela, menores os erros médios quadráticos.

Tabela 2.1: Erro médio quadrático da identificação do RS para a abordagem com método de janela única

Tamanho da janela	$EMQ_{00}$	$EMQ_{11}$
10	0,0143	0,0405
50	0,0052	0,0106
100	0,0041	0,0049

Na Seção 2.2.2 é apresentado o método de identificação do RS com o método de multi-janela, que oferece melhores resultados que o de janela única com comprimentos de janela menores.

### 2.2.2 Abordagem com multi-janela

Uma desvantagem da abordagem com janela única é que não há garantia de que o conteúdo da janela num dado instante de tempo sempre contém informação completa sobre o estado do RS. Assim o segundo e o terceiro caso de (2.44) podem ser usados com frequência quando  $l_w$  é pequeno. A solução de aumentar  $l_w$  não é interessante pois ela aumenta o *tempo de adaptação*, i.e. o tempo em que a janela observa a mudança entre dois processos estacionários  $u_{s-1}$  e  $u_s$ , e a estimativa do novo processo  $u_s$  é perturbada pelo antigo processo  $u_{s-1}$ . A abordagem com multi-janela é apresentada nesse contexto para manter a história prévia de cada estado separadamente. Um exemplo é mostrado na Figura 2.16.

O número de janelas é igual ao número de estados do RS. Cada janela  $W_i$  corresponde a um estado  $s_r = i$  e todas elas possuem comprimento  $l_w$ . As transições anteriores a partir do estado  $i$  são armazenadas na janela  $W_i$ . O *Buffer da Requisição Anterior* (BRA) armazena  $s_r(t-1)$  e controla o seletor de janela que seleciona a janela  $W_i$  tal que  $s_r(t-1) = i$ . Apenas essa janela selecionada realiza a operação de deslocamento. Então  $\hat{P}_{ij}$  pode ser definido com na Equação (2.47).

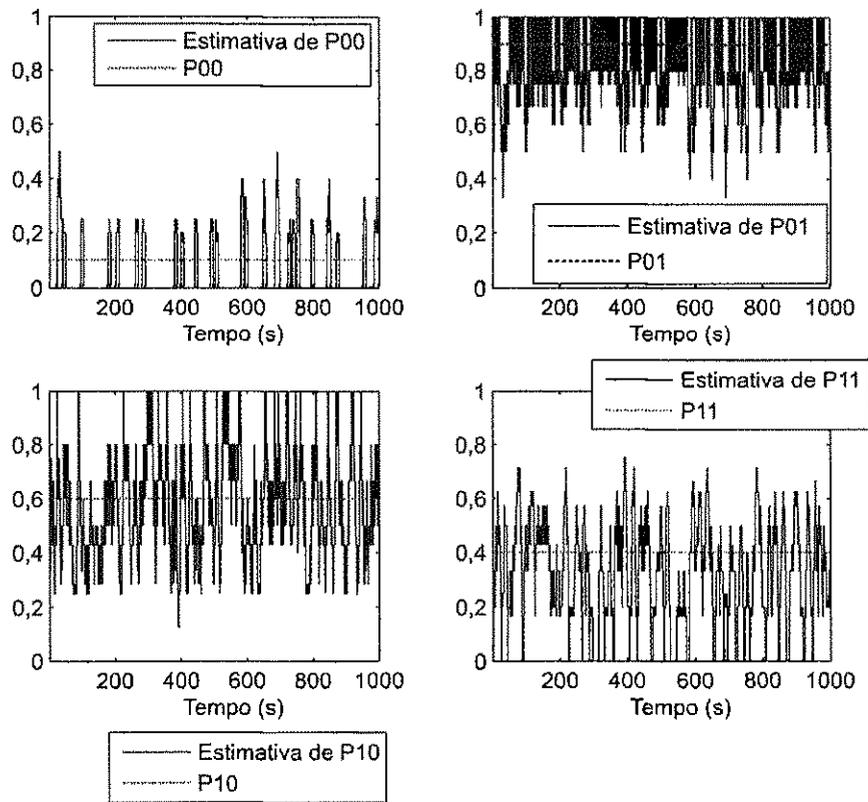


Figura 2.13: Identificação de um RS através do método de janela única com comprimento 10.

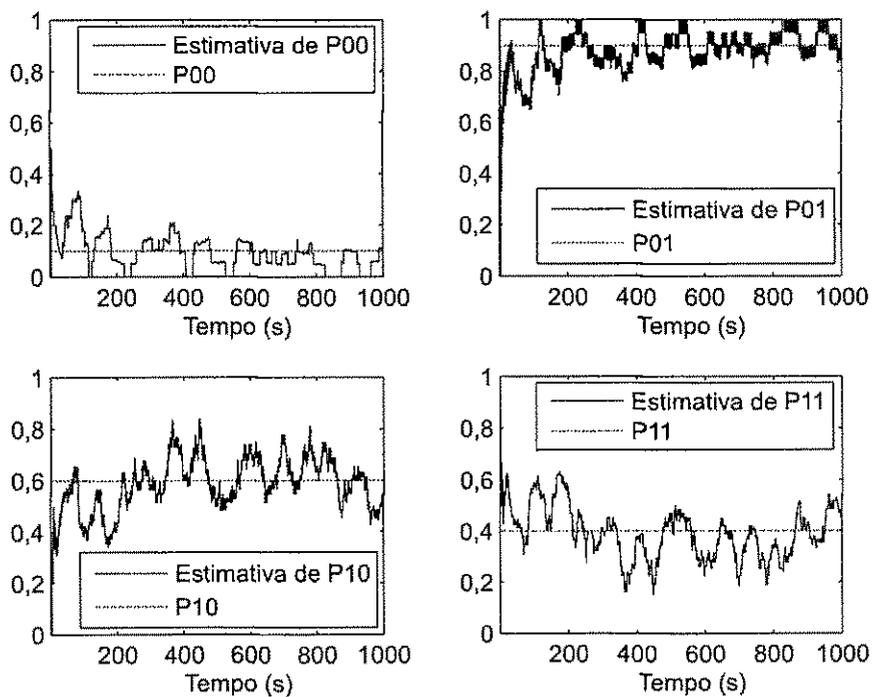


Figura 2.14: Identificação de um RS através do método de janela única com comprimento 50.

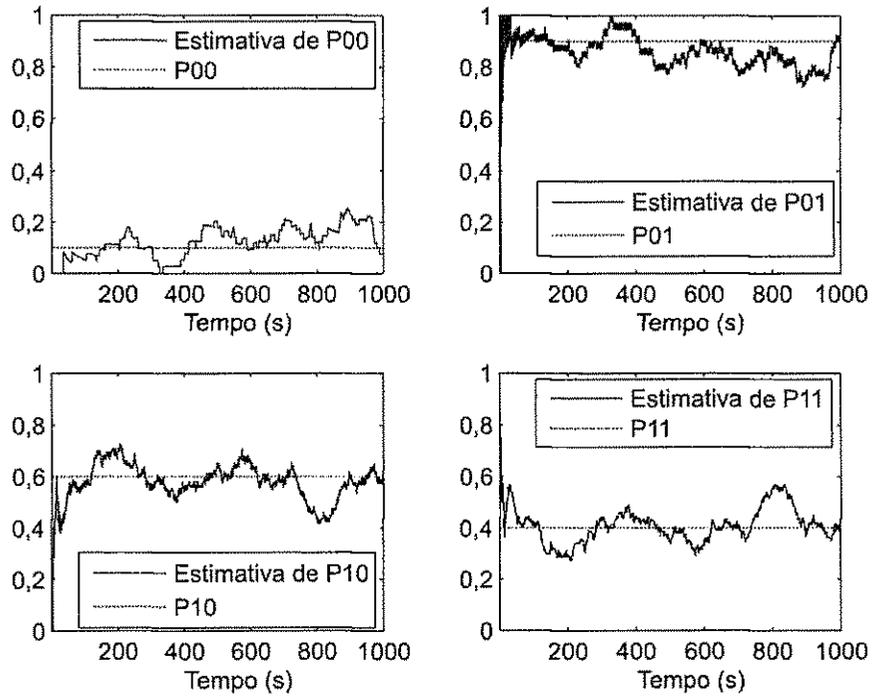


Figura 2.15: Identificação de um RS através do método de janela única com comprimento 100.

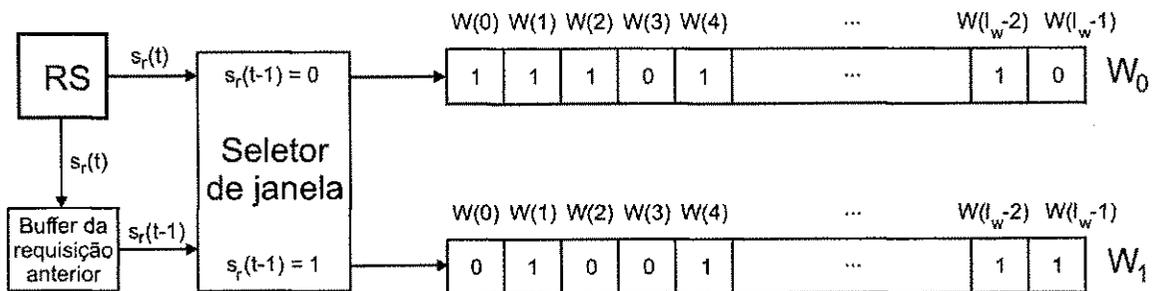


Figura 2.16: Exemplo de uso de multi-janela para RS de dois estados.

$$\hat{p}_{r_i, r_j}^{RS}(t) = \frac{\sum_{k=0}^{l_w-1} (W_i(k) = j)}{l_w} \text{ para todo } i, j. \quad (2.47)$$

**Exemplo 8** *Seja o RS com matriz de transição explicitada na Equação (2.48). Foi utilizado o método de identificação do RS com a abordagem de multi-janela através de três comprimentos de janela: 10, 50 e 100. Os resultados são mostrados nas Figuras 2.17, 2.18 e 2.19. Foi utilizado período de 1s e o tempo de simulação de 1000s. Em cada figura a linha tracejada representa a probabilidade real e a linha sólida é a identificação da probabilidade. Como esperado, os resultados oferecidos pela abordagem com multi-janela foram mais satisfatórios que os com janela-única. No entanto, uma desvantagem é o tempo de adaptação do método de multi-janela (aproximadamente o dobro do comprimento da janela), que é mais longo em comparação ao de janela única (aproximadamente o comprimento da janela).*

$$P^{RS} = \begin{matrix} & \begin{matrix} 0 & 1 \end{matrix} \\ \begin{matrix} 0 \\ 1 \end{matrix} & \begin{bmatrix} 0,1 & 0,9 \\ 0,6 & 0,4 \end{bmatrix} \end{matrix} \quad (2.48)$$

*Um comparativo quantitativo para o desempenho da estimação do RS através dos diferentes tamanhos de janela pode ser realizado através do cálculo do EMQ para as probabilidades  $p_{0,0}^{RS}(t)$  e  $p_{1,1}^{RS}(t)$ , conforme apresentado na Tabela 2.2. Percebe-se que, quanto maior o comprimento da janela, menores os erros médios quadráticos. Comparando os resultados das Tabelas 2.1 e 2.2, verifica-se que, apesar de ter um tempo de adaptação maior, o método de multi-janela apresenta menores erros médios quadráticos para o mesmo comprimento de janela.*

Tabela 2.2: Erro médio quadrático da identificação do RS para a abordagem com método de multi-janela

Tamanho da janela	$EMQ_{00}$	$EMQ_{11}$
10	0,0073	0,0221
50	0,0023	0,0100
100	0,0020	0,0088

### 2.2.3 Obtenção da política de energia

Uma vez obtidas as estimativas das probabilidades de transição do RS, é necessário obter uma política de energia apropriada para essas estimativas. A computação das técnicas de otimização introduzidas por Paleologo et al.(4) em tempo de execução é inviável, pois

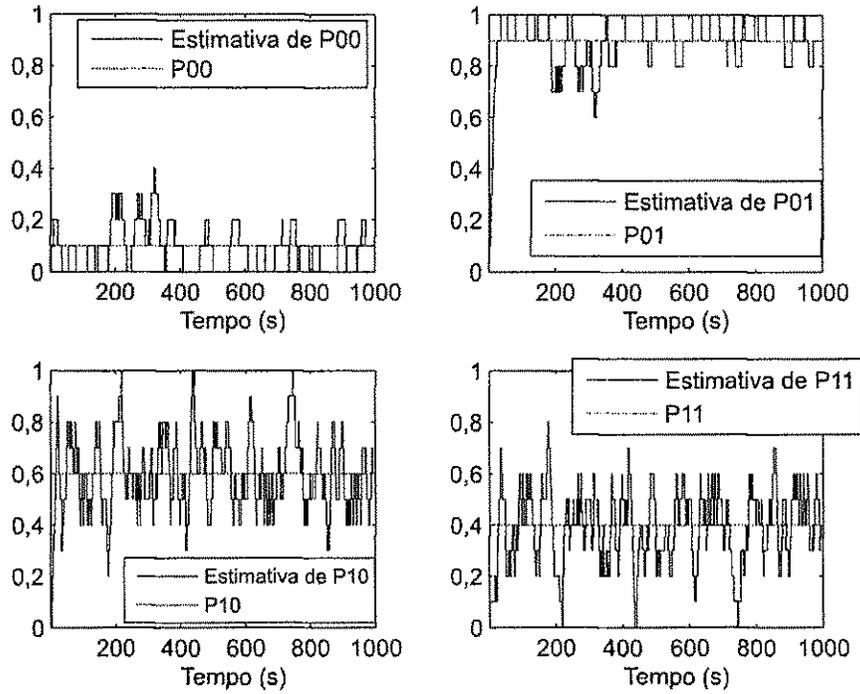


Figura 2.17: Identificação de um RS através do método de multi-janela com comprimento 10.

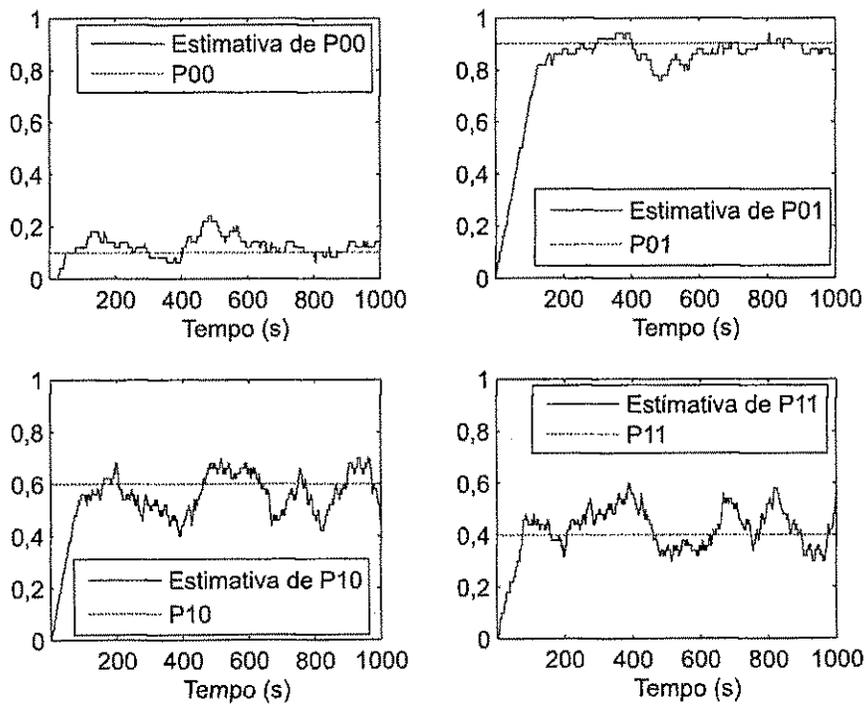


Figura 2.18: Identificação de um RS através do método de multi-janela com comprimento 50.

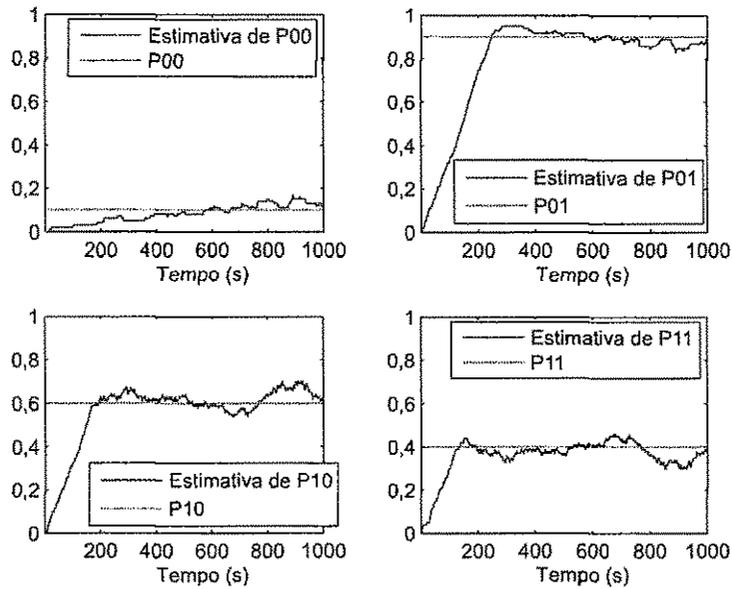


Figura 2.19: Identificação de um RS através do método de multi-janela com comprimento 100.

devido ao alto custo computacional de se recalcular uma política ótima a cada intervalo de tempo, as hipóteses de que o GE toma decisões rápidas com gasto de energia negligenciável podem ser violadas. Como alternativa para esse problema, Chung et al.(1) propõem um método de tabela de consulta (*look-up table*) com interpolação linear para a obtenção da política de energia.

Considerando um RS com dois estados, sua matriz de transição pode ser caracterizada pelas probabilidades de requisição de usuário  $R_i \in [0, 1]$ ,  $i = 0, 1$ . Cada dimensão dessa matriz é amostrada com um número finito de amostras, cada uma denotada por  $R_{ij}$ ,  $j = 0, 1, \dots, NS_i - 1$ , onde  $NS_i$  é o número de amostras para a dimensão  $i$ . Tal como ilustrado na Figura 2.20, é construída uma tabela de consulta denominada *tabela de políticas*.

A cada elemento da tabela de políticas corresponde um RS com probabilidades de requisições de usuário  $(R_{0j}, R_{1k})$ , ( $j = 0, 1, \dots, NS_0 - 1$  e  $k = 0, 1, \dots, NS_1 - 1$ ). Por sua vez, um elemento da tabela de políticas é uma tabela de decisão (ver exemplo 6), que é uma tabela com número de linhas igual ao número de estados do sistema e número de colunas igual ao de comandos passados pelo GE para o PS. Para cada par  $(R_{0j}, R_{1k})$  é realizada uma otimização de política de energia ainda na fase de projeto do GE.

Num dado intervalo de tempo, uma estimativa  $(\hat{R}_0, \hat{R}_1)$  é obtida e são escolhidos dois índices consecutivos para cada dimensão:  $R_{0j} \leq \hat{R}_0 \leq R_{0(j+1)}$  e  $R_{1k} \leq \hat{R}_1 \leq R_{1(k+1)}$ . O cálculo da tabela de decisão para o estado do sistema e o  $(\hat{R}_0, \hat{R}_1)$  utiliza quatro tabelas

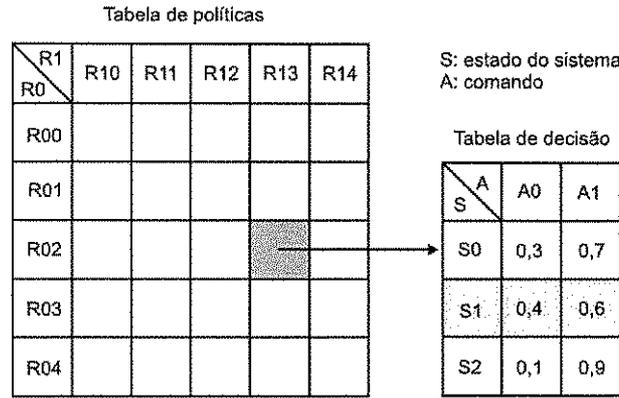


Figura 2.20: Exemplo de tabela de políticas com  $NS_0 = NS_1 = 5$ .

de decisão correspondentes a esses índices escolhidos, havendo em cada uma dessas uma linha associada ao estado atual do sistema, denotada por  $EA$ . Uma técnica de interpolação bi-dimensional é aplicada a essas quatro linhas para calcular-se a linha de decisão final. Seja  $f : \mathbb{R} \rightarrow \mathbb{R}$  uma função unidimensional, então é possível interpolar linearmente a imagem  $f(x)$  de qualquer ponto  $x$  entre dois outros,  $x_1$  e  $x_2$ , através da expressão (2.49). Para realizar a técnica de interpolação bi-dimensional, a de interpolação unidimensional apresentada em (2.49) é aplicada recursivamente. Nos casos em que  $\hat{R}_i > R_{i(NS_i-1)}$  ou  $\hat{R}_i < R_{i0}$ , deve ser utilizada extrapolação.

$$f(x) = \frac{(f(x_2) - f(x_1))x + x_2f(x_1) - x_1f(x_2)}{x_2 - x_1} \quad (2.49)$$

**Exemplo 9** Seja  $R_{0j} \in \{0, 0,25, 0,5, 0,75, 1\}$  e  $R_{1k} \in \{0, 0,25, 0,5, 0,75, 1\}$ . As estimativas de probabilidade de requisição são  $\hat{R}_0 = 1/3$  e  $\hat{R}_1 = 0,625$ . Assim  $R_{01} = 0,25$ ,  $R_{02} = 0,5$ ,  $R_{12} = 0,5$ ,  $R_{13} = 0,75$ , tal como ilustrado na Figura 2.21. Um exemplo do processo iterativo de interpolação é mostrado na Figura 2.22. Como exemplo, o cálculo da probabilidade referente ao comando A0 para  $(R_{01}, \hat{R}_1)$  é mostrado em (2.50), e o cálculo da probabilidade referente ao comando A0 para  $(\hat{R}_0, \hat{R}_1)$  é mostrado em (2.51).

$$\frac{(0,5 - 0,7) \cdot 0,625 + 0,75 \cdot 0,7 - 0,5 \cdot 0,5}{0,75 - 0,5} = 0,6 \quad (2.50)$$

$$\frac{(0,3 - 0,6) \cdot \frac{1}{3} + 0,5 \cdot 0,6 - 0,25 \cdot 0,3}{0,5 - 0,25} = 0,5 \quad (2.51)$$

**Exemplo 10** Considere o RS não estacionário em que as probabilidades de transição variam conforme (2.52) e (2.53), a FS do exemplo 2, e o PS e GE do exemplo 5. Deseja-se simular esse sistema durante 10000 passos de simulação, o que resulta num fator de desconto  $\xi = 1 - 1/10000 = 0,9999$ . Deseja-se também que o número médio de requisições

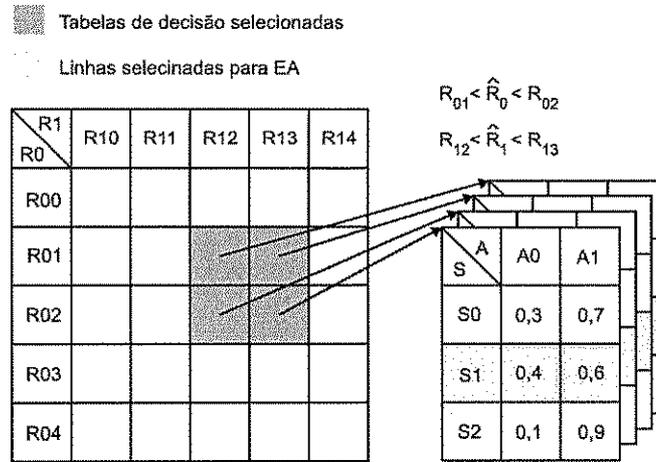


Figura 2.21: Tabela de decisão e exemplo de seleção de linhas da tabela de políticas.

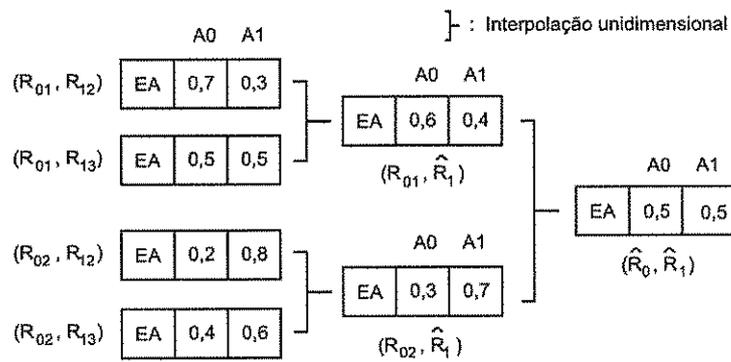


Figura 2.22: Exemplo de tabela de políticas com  $NS_0 = NS_1 = 5$ .

pendentes na FS seja de no máximo 50%, e que a perda de requisições média seja de no máximo 10%. O período de aplicação do GE é de 1s.

$$p_{0,0}^{RS}(t) = \begin{cases} 0,85, & \text{se } 0 \leq t \leq 1000s \\ 0,80, & \text{se } 1000s < t \leq 2000s \\ 0,75, & \text{se } 2000s < t \leq 3000s \\ 0,70, & \text{se } 3000s < t \leq 4000s \\ 0,25, & \text{se } 4000s < t \leq 5000s \\ 0,75, & \text{se } 5000s < t \leq 6000s \\ 0,85, & \text{se } 6000s < t \leq 7000s \\ 0,90, & \text{se } 7000s < t \leq 8000s \\ 0,85, & \text{se } 8000s < t \leq 9000s \\ 0,80, & \text{se } t > 9000s \end{cases} \quad (2.52)$$

$$p_{1,1}^{RS}(t) = \begin{cases} 0,20, & \text{se } 0 \leq t \leq 1000s \\ 0,25, & \text{se } 1000s < t \leq 2000s \\ 0,30, & \text{se } 2000s < t \leq 3000s \\ 0,35, & \text{se } 3000s < t \leq 4000s \\ 0,75, & \text{se } 4000s < t \leq 5000s \\ 0,30, & \text{se } 5000s < t \leq 6000s \\ 0,35, & \text{se } 6000s < t \leq 7000s \\ 0,25, & \text{se } 7000s < t \leq 8000s \\ 0,15, & \text{se } 8000s < t \leq 9000s \\ 0,10, & \text{se } t > 9000s \end{cases} \quad (2.53)$$

Para identificação do RS foi utilizado o método de multi-janela com comprimento 100. O resultado da identificação ao longo da simulação é mostrado na Figura 2.23.

Simulou-se também o gerenciamento de energia com as requisições definidas em (2.52) e (2.53) através a técnica de Paleologo et al.(4) (discutida na seção 2.1) para fins de comparação. Como explicado no exemplo 6, a política de energia para essa técnica foi otimizada para os valores de probabilidade de transição do RS definido em (2.52) e (2.53) entre 0 e 1000s.

Os resultados de simulação são apresentados na Tabela 2.3. Quatro métricas foram calculadas para avaliar a qualidade dos algoritmos:

- I: Corrente média (unidade: mA)
- $\hat{D}$ : Penalidade de desempenho média por período (%)
- $\hat{L}$ : Perda média de requisições por período (%)

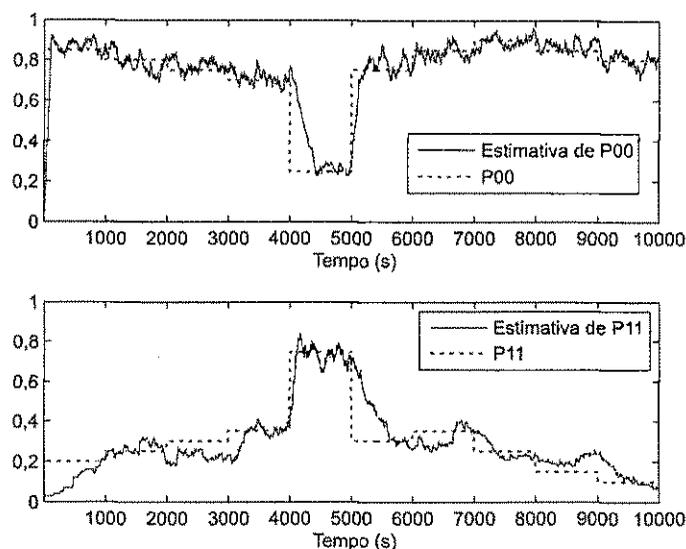


Figura 2.23: Identificação do RS definido em (2.52) e (2.53) através do método de multi-janela com comprimento 100.

Tabela 2.3: Comparação de algoritmos.

Algoritmo	$I(mA)$	$\hat{D}$	$\hat{L}$
Paleologo et al.(4)	116,15	56,44%	15,87%
Chung et al.(1)	129,7	46,72%	12,64%

O consumo médio da técnica de Chung et al.(1) foi maior que o de Paleologo et al.(4), porém a penalidade de desempenho média e a perda de requisições média foram menores. O consumo médio de corrente de ambas as técnicas foi menor que se o sistema permanecesse sempre ligado (200mA). A penalidade de desempenho média através da técnica de Paleologo et al.(4) foi maior que o valor máximo desejado de 50%, pois trata-se de uma técnica não-adaptativa que está submetida a um RS não-estacionário. A perda média de requisições para ambas as técnicas foram maiores que o máximo valor desejado de 10%, mas o apresentado pela técnica de Paleologo et al.(4) foi maior que pela técnica de Chung et al.(1). Verifica-se portanto que, no presente exemplo, uma maior corrente média pela técnica de Chung et al.(1) foi compensada por menores penalidade de desempenho média e perda de requisições média pois, dado que o RS não estacionário muda as suas probabilidades de transição ao longo do tempo e essa técnica identifica o RS em tempo de execução, ela pode aplicar em cada instante de tempo uma política de energia mais próxima da ótima para as probabilidades de transição reais do RS em cada instante de tempo. Já a técnica de Paleologo et al.(4) utiliza uma política de energia somente otimizada para as probabilidades de transição do RS entre 0 e 1000 iterações. Em síntese, para um RS não estacionário a técnica de Chung et al.(1) tem melhor desempenho que a de Paleologo

*et al.(4) pois a primeira é adaptativa enquanto a segunda é não-adaptativa.*

## 2.3 Conclusões

É importante perceber que resultados ótimos globais não podem ser provados para a técnica de Chung et al.(1) para requisições de serviço não-estacionárias, sendo essa uma técnica heurística. Como demonstrado no exemplo 10 a eficiência dessa técnica só pode ser evidenciada através de simulações ou experimentos.

As soluções propostas por Paleologo et al.(4) e Chung et al.(1) são técnicas de otimização de uso de energia, que podem ser bem aplicadas a um modelo linear de bateria. A característica de taxa de capacidade e o efeito de recuperação não são considerados em modelos lineares de bateria. Então as técnicas de Paleologo et al.(4) e Chung et al.(1) podem não otimizar a autonomia de baterias de dispositivos reais alimentados a bateria.

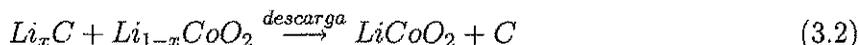
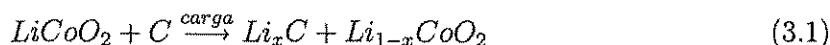
As propriedades do processo de descarga de baterias a serem usados pela técnica de GDE orientada à autonomia da bateria proposta na Seção 4.2 são descritas na Seção 3.2 usando o modelo introduzido por Rakhmatov e Vrudhula(12).

## Capítulo 3

### Modelos de bateria

Uma bateria é constituída de um eletrodo positivo denominado *cátodo* e um eletrodo negativo denominado *ânodo* separados por um *eletrólito*. Para fornecer corrente elétrica a um circuito externo, o ânodo libera elétrons para o circuito enquanto o cátodo recebe elétrons do circuito. Tais elétrons originam-se de reações químicas entre os eletrodos e íons presentes no eletrólito denominados *espécies eletroativas* (12).

Numa bateria de Lítio-Íon (20), o lítio pode ser extraído e depositado tanto no ânodo quanto no cátodo. Na descarga, o lítio é extraído do ânodo e inserido no cátodo. No processo de carga, o lítio é extraído do cátodo e depositado no ânodo. Uma célula de Lítio-Íon de 3,7V é composta de um ânodo de carbono (*C*) e um cátodo de óxido de metal (*LiCoO<sub>2</sub>*). A reação química entre esses componentes é representada nas Equações (3.1) e (3.2), em que *x* é o percentual de *Li*. É importante destacar que os íons de lítio não são oxidados, são somente transportados do ânodo para o cátodo ou vice-versa. O cátodo é oxidado no processo de carga, e reduzido na descarga.



A Figura 3.1(a) representa as vizinhanças de um eletrodo de uma bateria em estado carregado, situação em que as espécies eletroativas estão uniformemente distribuídas. Apesar de uma bateria ser constituída de dois eletrodos, na Figura 3.1 só é ilustrado um eletrodo por simplicidade, e, assim, o exposto a seguir é aplicado a ambos os eletrodos de uma bateria. A conexão de uma carga aos terminais da bateria inicia o processo de descarga e estabelece um fluxo externo de elétrons entre os eletrodos. Reações eletroquímicas causam a redução do número de espécies eletroativas próximas ao eletrodo. Um gradiente de concentração de espécies não nulo ao longo do eletrólito (ver Figura 3.1(b)) se forma e é iniciada a difusão de espécies ao longo do eletrólito em direção ao eletrodo para

reequilibrar a concentração de espécies. Se a carga é desligada, a corrente é anulada, e devido à difusão de espécies, a concentração próxima à superfície do eletrodo aumenta até o gradiente de concentração se anular (ver Figura 3.1(c)). Diz-se que houve a recuperação. A concentração de espécies fica novamente uniformemente distribuída ao longo do eletrólito, mas com valor menor. Quando a concentração de espécies eletroativas na superfície do eletrodo é inferior a um determinado limite, considera-se a bateria descarregada, pois as reações químicas não podem mais ser sustentadas (ver Figura 3.1(d)).

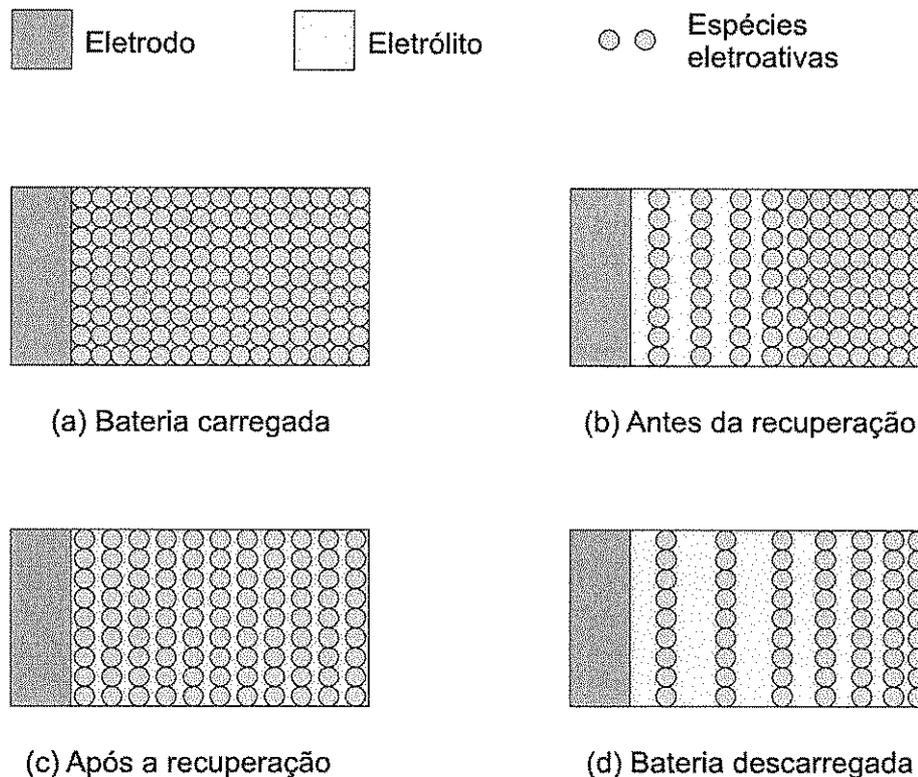


Figura 3.1: Efeito de recuperação na bateria.

Há três importantes valores de tensão numa bateria (21): (i) a *tensão de circuito aberto* ( $V_{ca}$ ), que é o valor inicial da tensão nos terminais da bateria totalmente carregada e em circuito aberto; (ii) a *tensão de operação* ( $V$ ) que é a tensão da bateria em circuito fechado; (iii) a *tensão de corte* ( $V_{co}$ ), que é a tensão na qual considera-se a bateria descarregada. A capacidade teórica de uma bateria (expressa em  $Ah$ ) é especificada pela quantidade de materiais ativos nela presente (20).

Modelos eletroquímicos de baterias, como o introduzido por Newman et al. (22), contemplam tanto aspectos macroscópicos, como tensão e corrente, quanto aspectos microscópicos como distribuição de concentração. A sua formulação em equações diferenciais parciais é complexo e demanda considerável tempo de simulação (dias), além de que a caracterização da bateria requer informações específicas, muitas vezes indisponíveis por

serem tecnologia proprietária.

Chen e Rincón-Mora(23) desenvolveram um modelo elétrico acurado para bateria que contempla tensão de circuito aberto não-linear, auto-descarga e dependências com a temperatura, número de ciclos, corrente e tempo de armazenamento. A caracterização rigorosa dos componentes desse modelo elétrico de bateria pode ser demasiadamente complexa e longa, levando por exemplo a até centenas de ensaios variando-se somente a temperatura. O modelo é simplificado ignorando-se auto-descarga, número de ciclos e temperatura. Contudo a caracterização desse modelo elétrico de bateria, mesmo em sua versão simplificada, foge ao escopo desse trabalho.

Panigrahi et al.(24) introduziram um modelo estocástico de bateria que contempla tanto a característica de taxa de capacidade quanto o efeito de recuperação. Para isso o comportamento da bateria é modelado como um processo estocástico discreto no tempo em que o estado do processo representa o estado de carga da bateria. Apesar desse modelo ser mais rápido e poder ter acurácia próxima a um modelo de equações diferenciais parciais do compartimento eletroquímico da bateria, Panigrahi et al.(24) não mostram como seu modelo pode ser usado por um algoritmo de otimização da autonomia da bateria, apenas utilizado-o de forma heurística.

Rong e Pedram(2) desenvolvem um modelo estocástico de bateria baseado em cadeias de Markov no tempo contínuo, mas formulam o problema de otimização da política de energia em função da minimização do consumo de energia, o que não implica em otimização da autonomia da bateria.

Dadas as deficiências apresentadas pelos modelos de bateria anteriormente apresentados, verifica-se a necessidade da utilização de um modelo de bateria que: (i) represente tanto a característica de taxa de capacidade quanto o efeito de recuperação; (ii) seja acurado e ao mesmo tempo mais rápido que um modelo de equações diferenciais parciais do comportamento eletroquímico da bateria; (iii) ofereça ferramentas para auxiliar o problema de maximização da autonomia da bateria. O modelo apresentado na seção 3.1 possui essas vantagens e será utilizado nesse trabalho para derivar-se a política de gerenciamento de energia proposta.

### 3.1 Modelo Analítico

A autonomia da bateria é o instante  $t = L$  em que a concentração de espécies eletroativas na superfície do eletrodo cai além de um dado nível de corte. Rakhmatov e Vrudhula(12) demonstram que o tempo de falha  $L$  da bateria (autonomia) se relaciona à corrente de descarga  $i(t)$  conforme a Equação (3.3), em que  $\alpha$  e  $\beta$  são dois parâmetros da bateria a serem estimados.

$$\alpha = \int_0^L i(\tau) d\tau + 2 \sum_{m=1}^{\infty} \int_0^L i(\tau) e^{-\beta^2 m^2 (L-\tau)} d\tau \quad (3.3)$$

A unidade de  $\alpha$  é coulombs e a de  $\beta^2$  é  $s^{-\frac{1}{2}}$ . O significado do lado direito da Equação (3.3) é a capacidade da bateria. O primeiro termo é a carga total que a bateria fornece para o sistema externo. O segundo termo é a quantidade de carga que a bateria não pôde oferecer ao sistema pois não estava disponível na superfície do eletrodo quando no tempo de falha.

Numa bateria ideal, toda a sua capacidade é oferecida ao sistema. Assim quanto mais a bateria se aproxima de uma fonte ideal, maior o  $\beta$ , pois o segundo termo do lado direito da Equação (3.3) diminui. Isso ocorre porque quanto maior o  $\beta$ , mais rápida é a difusão, e as espécies eletroativas conseguem alcançar a superfície do eletrodo mais rapidamente, fornecendo energia na taxa demandada pelo sistema.

Quanto menor o  $\beta$ , mais a bateria se distancia de uma bateria ideal. No tempo de falha só uma parte da capacidade da bateria foi fornecida para o sistema, e se houver um período em que a corrente se anula, a concentração de espécies eletroativas na superfície do eletrodo pode aumentar por difusão. Assim, torna-se disponível para o sistema parte da carga que antes era indisponível.

Para estimar os parâmetros  $\alpha$  e  $\beta$  é necessário realizar uma série de ensaios com corrente constante, caso em que a Equação (3.3) é simplificada para a Equação (3.4).

$$\alpha = IL \left[ 1 + 2 \sum_{m=1}^{\infty} \frac{1 - e^{-\beta^2 m^2 L}}{\beta^2 m^2 L} \right] \quad (3.4)$$

Para cada ensaio é aplicada uma corrente constante  $I_{(k)}$  e medido um tempo de falha  $L_{(k)}$ . Dado o conjunto de correntes  $\{I_{(1)}, \dots, I_{(N)}\}$  e tempos de falha  $\{L_{(1)}, \dots, L_{(N)}\}$ ,  $\alpha$  e  $\beta$  são estimados minimizando a soma de quadrados  $\sum |I_{(k)} - \hat{I}_{(k)}|^2$ , em que  $\hat{I}_{(k)}$  é a estimativa da corrente que leva a bateria a extinguir-se no tempo  $L_{(k)}$  segundo a Equação (3.5).

$$\hat{I}_{(k)} = \frac{\alpha}{L_{(k)} + 2 \sum_{m=1}^{\infty} \frac{1 - e^{-\beta^2 m^2 L_{(k)}}}{\beta^2 m^2}} \quad (3.5)$$

**Exemplo 11** A uma bateria de Lítio-Íon foram aplicadas várias correntes constantes e para cada uma foi obtido um tempo de falha (autonomia), conforme a Tabela 3.1. A tensão de circuito aberto da bateria é de 4,2V e a de corte é de 3,0V.

A partir dos dados da Tabela 3.1, os valores estimados dos parâmetros da bateria são  $\alpha = 39593 \text{mA} \cdot \text{min}$  e  $\beta = 0,557 \text{min}^{-\frac{1}{2}}$ . A Figura 3.2 mostra como a curva de corrente versus autonomia da bateria traçada a partir dos parâmetros estimados se aproxima da curva traçada com os valores da Tabela 3.1.

$I_{(k)}$ (mA)	$L_{(k)}$ (min)
120	300
220	180
320	120
420	80
520	65
615	55
715	45
815	40
910	32
1010	30

Tabela 3.1: Autonomias de uma bateria para diferentes correntes constantes.

Seja o perfil de descarga da bateria uma sequência de tarefas cada uma impondo uma corrente constante à bateria, como ilustrado na Figura 3.3, em que  $I_k$ ,  $\Delta_k$  e  $t_k$  são a corrente, duração e instante de início da tarefa  $k$ , respectivamente. Seja o conjunto de correntes  $S_I = \{I_k | k = 0, 1, \dots, n - 1\}$ , o de durações  $S_\Delta = \{\Delta_k | k = 0, 1, \dots, n - 1\}$ , e o de instantes iniciais  $S_t = \{t_k | k = 0, 1, \dots, n - 1\}$ . Assumindo que a bateria atinge seu tempo de falha durante a tarefa  $u$ , a aplicação da Equação (3.3) a esse perfil de descarga resulta em (3.6) que pode ser simplificada para (3.10) removendo os valores de correntes constantes das integrais.

$$\alpha = \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k d\tau \right] + \int_{t_u}^L I_u d\tau + 2 \sum_{m=1}^{\infty} \left\{ \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k e^{-\beta^2 m^2 (L-\tau)} d\tau \right] + \int_{t_u}^L I_u e^{-\beta^2 m^2 (L-\tau)} d\tau \right\} \Rightarrow \quad (3.6)$$

$$\Rightarrow \alpha = \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k d\tau \right] + \int_{t_u}^L I_u d\tau + 2 \sum_{m=1}^{\infty} \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k e^{-\beta^2 m^2 (L-\tau)} d\tau \right] + 2 \sum_{m=1}^{\infty} \int_{t_u}^L I_u e^{-\beta^2 m^2 (L-\tau)} d\tau \Rightarrow \quad (3.7)$$

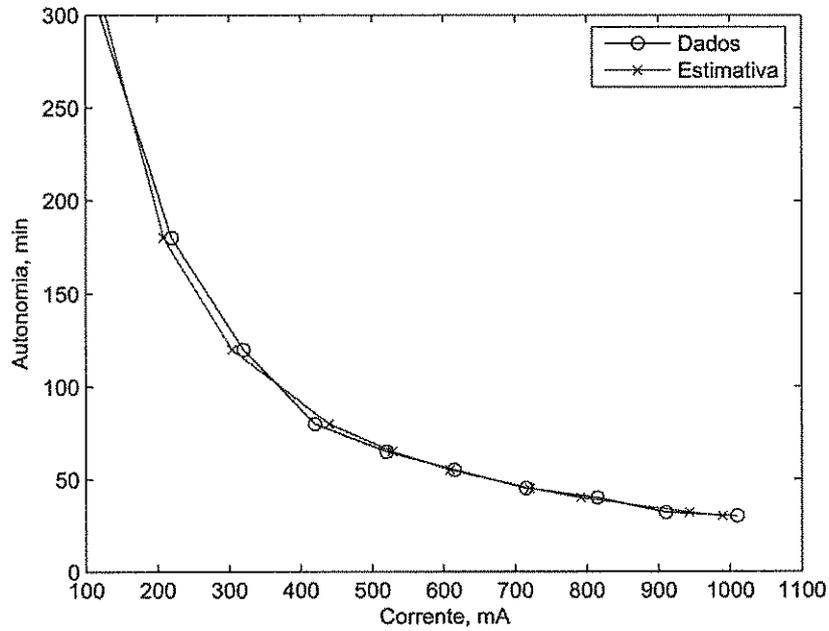


Figura 3.2: Autonomia da bateria em função de corrente de descarga constante.

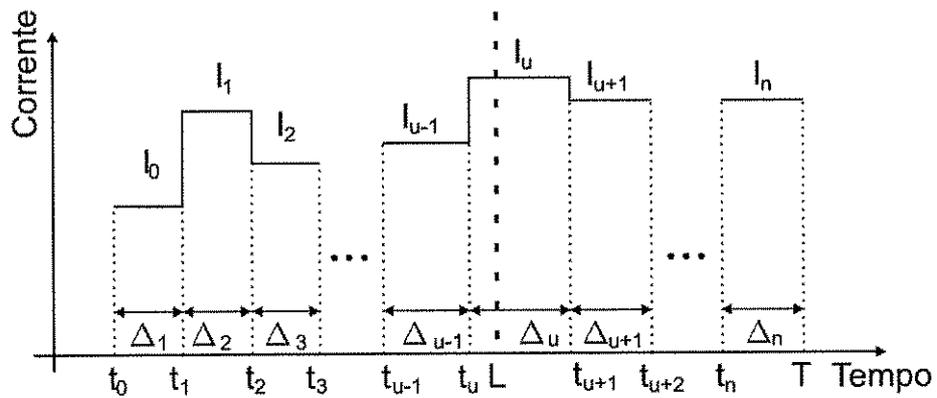


Figura 3.3: Perfil de descarga para a bateria.

$$\begin{aligned} \Rightarrow \alpha &= \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k d\tau \right] + \sum_{k=0}^{u-1} 2 \sum_{m=1}^{\infty} \left[ \int_{t_k}^{t_k+\Delta_k} I_k e^{-\beta^2 m^2 (L-\tau)} d\tau \right] + \\ &+ \int_{t_u}^L I_u d\tau + 2 \sum_{m=1}^{\infty} \int_{t_u}^L I_u e^{-\beta^2 m^2 (L-\tau)} d\tau \Rightarrow \end{aligned} \quad (3.8)$$

$$\begin{aligned} \Rightarrow \alpha &= \sum_{k=0}^{u-1} \left[ \int_{t_k}^{t_k+\Delta_k} I_k d\tau + 2 \sum_{m=1}^{\infty} \int_{t_k}^{t_k+\Delta_k} I_k e^{-\beta^2 m^2 (L-\tau)} d\tau \right] + \\ &+ \int_{t_u}^L I_u d\tau + 2 \sum_{m=1}^{\infty} \int_{t_u}^L I_u e^{-\beta^2 m^2 (L-\tau)} d\tau \Rightarrow \end{aligned} \quad (3.9)$$

$$\begin{aligned} \Rightarrow \alpha &= \sum_{k=0}^{u-1} I_k \left[ \int_{t_k}^{t_k+\Delta_k} d\tau + 2 \sum_{m=1}^{\infty} \int_{t_k}^{t_k+\Delta_k} e^{-\beta^2 m^2 (L-\tau)} d\tau \right] + \\ &+ I_u \left[ \int_{t_u}^L d\tau + 2 \sum_{m=1}^{\infty} \int_{t_u}^L e^{-\beta^2 m^2 (L-\tau)} d\tau \right] \end{aligned} \quad (3.10)$$

Resolvendo as integrais em (3.10), temos (3.11) e (3.12).

$$\int_{t_i}^{t_f} d\tau = t_f - t_i \quad (3.11)$$

$$\begin{aligned} \int_{t_i}^{t_f} e^{-\beta^2 m^2 (t-\tau)} d\tau &= e^{-\beta^2 m^2 t} \int_{t_i}^{t_f} e^{\beta^2 m^2 \tau} d\tau = e^{-\beta^2 m^2 t} \left\{ \frac{e^{\beta^2 m^2 \tau}}{\beta^2 m^2} \Big|_{t_i}^{t_f} \right\} = \\ &= e^{-\beta^2 m^2 t} \left[ \frac{e^{\beta^2 m^2 t_f} - e^{\beta^2 m^2 t_i}}{\beta^2 m^2} \right] = \frac{e^{-\beta^2 m^2 (t-t_f)} - e^{-\beta^2 m^2 (t-t_i)}}{\beta^2 m^2} \end{aligned} \quad (3.12)$$

Seja  $F$  a função definida na Equação (3.13). Usando os resultados (3.11) e (3.12),  $F$  é simplificada para (3.14).

$$F(t, t_i, t_f, \beta) = \int_{t_i}^{t_f} d\tau + 2 \sum_{m=1}^{\infty} \int_{t_i}^{t_f} e^{-\beta^2 m^2 (t-\tau)} d\tau \quad (3.13)$$

$$F(t, t_i, t_f, \beta) = t_f - t_i + 2 \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (t-t_f)} - e^{-\beta^2 m^2 (t-t_i)}}{\beta^2 m^2} \quad (3.14)$$

Usando a função  $F$ , a Equação (3.10) é simplificada para a Equação (3.15).

$$\alpha = \sum_{k=0}^{u-1} I_k F(L, t_k, t_k + \Delta_k, \beta) + I_u F(L, t_u, L, \beta) \quad (3.15)$$

**Exemplo 12** A bateria caracterizada no exemplo 11 foi submetida a três perfis de descarga diferentes:

- Perfil D1, ilustrada na Figura 3.4(a): corrente constante de 800mA ao longo de toda a descarga;
- Perfil D2, ilustrada na Figura 3.4(b): a corrente é de 800mA durante 25 minutos, 400mA durante 10 minutos, e 800mA até exaurir a bateria;
- Perfil D3, ilustrada na Figura 3.4(c): durante 25 minutos a corrente é de 800mA, sendo anulada durante 10 minutos e novamente aplicada até exaurir a bateria;

Na Figura 3.5 é mostrada a carga residual da bateria para cada perfil de descarga. Para o perfil D1, a corrente é constante e assim a autonomia da bateria é a menor entre os três perfis. Para o perfil D2, quando ocorre uma redução de corrente, mesmo que essa não seja nula como no perfil D3, ocorre o efeito de recuperação e autonomia da bateria é maior que o perfil D1. Para o perfil D3, quando a corrente é anulada, ocorre o efeito de recuperação e a carga residual da bateria aumenta, assim estendendo a autonomia da bateria, que é a maior entre os três perfis.

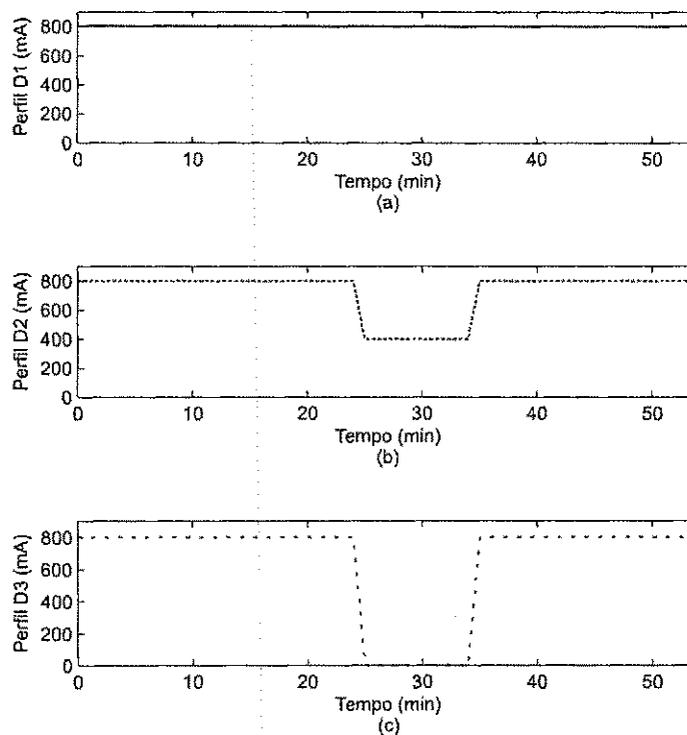


Figura 3.4: Perfis de descarga.

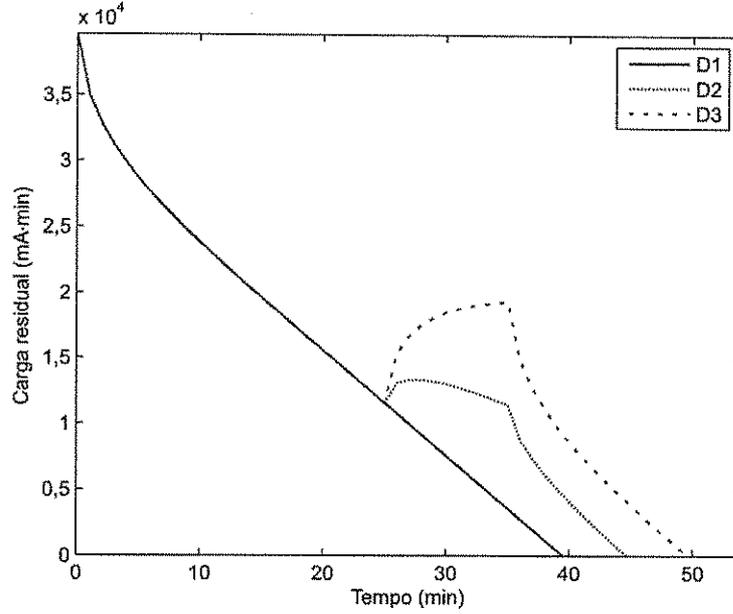


Figura 3.5: Carga residual da bateria para cada perfil de descarga da Figura 3.4.

### 3.2 Descarga da bateria e tempo de recuperação

Para o perfil de descarga de bateria ilustrado na Figura 3.3, Rakhmatov e Vrudhula(12) definem a função de custo de bateria  $\sigma(T)$  (Equação (3.16)) a ser minimizada para maximizar a autonomia da bateria. A função  $\sigma(T)$  representa a carga que a bateria perdeu até o instante  $T$ . Assim enquanto  $\sigma(T) < \alpha$  a bateria ainda não foi exaurida, o que só acontece quando  $\sigma(T) = \alpha$ .

$$\sigma(T) = \sum_{k=0}^{n-1} I_k F(T, t_k, t_k + \Delta_k, \beta) \quad (3.16)$$

Repondo a função  $F$  da Equação (3.14) em (3.16), é obtida a Equação (3.17).

$$\sigma(T) = \underbrace{\sum_{k=0}^{n-1} I_k \Delta_k}_{l(T)} + \underbrace{\sum_{k=0}^{n-1} I_k \left( 2 \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (T-t_k-\Delta_k)} - e^{-\beta^2 m^2 (T-t_k)}}{\beta^2 m^2} \right)}_{u(T)} \quad (3.17)$$

Como destacado por Rao e Vrudhula(25), o termo  $l(T)$  de (3.17) é a *carga perdida real*, ou seja, a que a bateria fornece para o circuito externo, e o termo  $u(T)$  é interpretado como a *carga indisponível* pois não estava disponível na superfície do eletrodo no instante  $T$ . Então  $a(T) = \alpha - \sigma(T)$  é a *carga residual* da bateria, enquanto  $a_L(T) = \alpha - l(T)$  é a carga residual se a bateria fosse ideal. Dado que  $u(T) \geq 0, \forall T \geq 0$ , é demonstrado em (3.18) que  $a(T) \leq a_L(T)$ , i.e. a carga residual da bateria é menor ou igual à carga residual se a bateria fosse ideal.

$$\begin{aligned}
 u(T) \geq 0 &\Rightarrow l(T) + u(T) \geq l(T) \Rightarrow \sigma(T) \geq l(T) \Rightarrow -\sigma(T) \leq -l(T) \Rightarrow \\
 &\Rightarrow \alpha - \sigma(T) \leq \alpha - l(T) \Rightarrow a(T) \leq a_L(T)
 \end{aligned}
 \tag{3.18}$$

**Exemplo 13** Um exemplo de processo de descarga é mostrado na Figura 3.6 para uma bateria de parâmetros  $\alpha = 60000 \text{mA} \cdot \text{min}$  e  $\beta = 0,35 \text{min}^{-\frac{1}{2}}$ . Para os primeiros 20 minutos, a corrente de descarga foi de 800 mA. A corrente é zero durante 20 minutos, e depois retorna a 800 mA até que a bateria seja completamente descarregada.

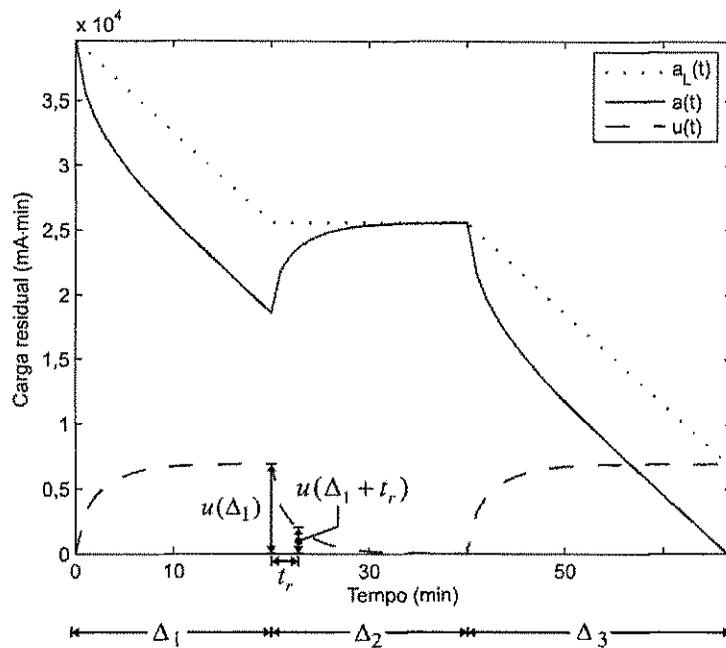


Figura 3.6: Exemplo de um processo de descarga.

Verifica-se que, no intervalo de tempo  $\Delta_2$  (veja a Figura 3.6),  $a(T)$  se aproxima de  $a_L(T)$  devido ao efeito de recuperação, pois a corrente é zero nesse intervalo de tempo.

É possível verificar que a diferença entre a carga residual se a bateria fosse ideal  $a_L(T)$  e a carga residual  $a(T)$  é igual à carga indisponível  $u(T)$ , como mostrado em 3.19.

$$a_L(T) - a(T) = \alpha - l(T) - \alpha + \sigma(T) = \sigma(T) - l(T) = u(T)
 \tag{3.19}$$

O efeito de recuperação diminui a carga indisponível, pois, por difusão, as espécies eletroativas atingem a superfície do eletrodo e se tornam disponíveis para novas reações químicas. Então deseja-se que, quando a corrente é zero, o tempo de inatividade seja tão longo quanto possível para que  $u(T)$  se aproxime de zero tanto quanto possível. O tempo de recuperação  $t_r(\epsilon)$  da série de cargas é definido por Rao e Vrudhula(25) como o

tempo adicional após o final da série de cargas tal que a carga indisponível alcança uma fração  $0 < \varepsilon < 1$  do seu valor do final da série de cargas. Por exemplo, considerando a série de cargas  $(I_1, \Delta_1)$  mostrada na Figura 3.6,  $t_r(\varepsilon)$  é o tempo adicional após  $\Delta_1$  para que  $u(t_r + \Delta_1) = \varepsilon u(\Delta_1)$ . Seja  $G$  a função definida em (3.20). Para uma série de cargas arbitrárias com  $\Delta = \sum_{k=0}^{n-1} \Delta_k$ , o tempo de recuperação é definido por (3.21).

$$G(t, t_i, t_f, \beta) = 2 \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (t-t_f)} - e^{-\beta^2 m^2 (t-t_i)}}{\beta^2 m^2} \quad (3.20)$$

$$\frac{u(t_r + \Delta)}{u(\Delta)} = \frac{\sum_{k=0}^{n-1} I_k G(t_r + \Delta, t_k, t_k + \Delta_k, \beta)}{\sum_{k=0}^{n-1} I_k G(\Delta, t_k, t_k + \Delta_k, \beta)} = \varepsilon \quad (3.21)$$

Apesar da dificuldade de encontrar uma expressão fechada para  $t_r(\varepsilon)$  a partir da Equação (3.21), o limitante superior expresso em (3.22) pode ser obtido (25).

$$\bar{t}_r(\varepsilon) = -\frac{1}{\beta^2} \log(1,5\varepsilon), \quad 0 < \varepsilon < 0,4 \quad (3.22)$$

Para o caso em que a bateria é submetido a uma seqüência de correntes  $I_k$ ,  $k = 1, \dots, n-1$  e depois ocorre uma corrente não nula  $I_n < I_{n-1}$ , como é o caso do perfil de descarga D2 do exemplo 12, também ocorre o efeito de recuperação, mas a corrente  $I_n$  exerce influência sobre a recuperação da carga indisponível, que é expressada em (3.23). Para  $t_r \rightarrow \infty$ ,  $u(t_r + \Delta)$  tende a uma constante devido ao efeito da corrente  $I_n$ , como mostrado em (3.24). Assim, o efeito de recuperação se deve ao efeito das correntes  $I_k$ ,  $k = 1, \dots, n-1$  sobre  $u(t_r + \Delta)$ , do qual deve-se descontar o efeito da corrente  $I_n$ , o que é expresso em (3.25). É possível então utilizar o mesmo limitante superior expresso em (3.22).

$$\begin{aligned} u(t_r + \Delta) &= I_n G(t_r + \Delta, \Delta, \Delta + t_r, \beta) + \sum_{k=0}^{n-1} I_k G(t_r + \Delta, t_k, t_k + \Delta_k, \beta) \\ &= I_n \cdot 2 \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (t_r + \Delta - \Delta - t_r)} - e^{-\beta^2 m^2 (t_r + \Delta - \Delta)}}{\beta^2 m^2} + \sum_{k=0}^{n-1} I_k G(t_r + \Delta, t_k, t_k + \Delta_k, \beta) \\ &= I_n \cdot 2 \sum_{m=1}^{\infty} \frac{1 - e^{-\beta^2 m^2 t_r}}{\beta^2 m^2} + \\ &+ \sum_{k=0}^{n-1} I_k \cdot 2 \sum_{m=1}^{\infty} \frac{e^{-\beta^2 m^2 (t_r + \Delta - t_k - \Delta_k)} - e^{-\beta^2 m^2 (t_r + \Delta - t_k)}}{\beta^2 m^2} \end{aligned} \quad (3.23)$$

$$\lim_{t_r \rightarrow \infty} u(t_r + \Delta) = I_n \cdot 2 \sum_{m=1}^{\infty} \frac{1}{\beta^2 m^2} = \frac{2I_n}{\beta^2} \sum_{m=1}^{\infty} \frac{1}{m^2} = \frac{2I_n}{\beta^2} \cdot \frac{\pi^2}{6} = \frac{\pi^2 I_n}{3\beta^2} \quad (3.24)$$

$$\frac{u(t_r + \Delta) - I_n G(t_r + \Delta, \Delta, \Delta + t_r, \beta)}{u(\Delta)} = \frac{\sum_{k=0}^{n-1} I_k G(t_r + \Delta, t_k, t_k + \Delta_k, \beta)}{\sum_{k=0}^{n-1} I_k G(\Delta, t_k, t_k + \Delta_k, \beta)} = \varepsilon \quad (3.25)$$

### 3.3 Conclusões

O modelo analítico de bateria de Rakhmatov e Vrudhula(12) oferece a vantagem de ser acurado e ao mesmo tempo mais rápido que um modelo de equações diferenciais parciais do comportamento eletroquímico da bateria, e também oferecer uma função custo para auxiliar o problema de maximização da autonomia da bateria. Esse modelo é capaz de representar tanto a característica de taxa de capacidade quanto o efeito de recuperação, sendo este último o de maior interesse para aumentar a autonomia da bateria.

Para uma política de gerenciamento de energia, escolher quanto tempo o sistema permanece desligado é um problema chave. Deseja-se que esse tempo seja tão longo quanto possível, para aproveitar o efeito de recuperação e assim aumentar a autonomia da bateria. Contudo, tempos maiores de desligamento podem reduzir o desempenho do sistema. Assim, a determinação do limitante superior para o tempo de recuperação da bateria é de grande importância, pois permite que política de gerenciamento de energia beneficie do efeito de recuperação tanto quanto se deseje sem prejudicar o desempenho.

Na Seção 4.2 é mostrado como a abordagem de GDE de Chung et al.(1) pode beneficiar do modelo de bateria de Rakhmatov e Vrudhula(12) usando políticas otimizadas para restrições de baixo desempenho em intervalos inativos, estendendo os intervalos de tempo em que a bateria se “recupera”, o que consideravelmente aumenta a autonomia da bateria.

## Capítulo 4

# Gerenciamento dinâmico de energia orientado à autonomia da bateria

A maioria das técnicas de gerenciamento dinâmico de energia tem como foco a minimização da energia consumida pelo sistema. Como já discutido, a minimização da energia consumida nem sempre implica na maior autonomia possível para um sistema alimentado a bateria. Surge assim a necessidade de avaliar formas de realizar gerenciamento dinâmico de energia que levem em conta modelos acurados da bateria e objetivem a maximização da autonomia da bateria.

Rong e Pedram(2) modelam uma bateria como um PDMTC estacionário e a incluem no modelo do sistema em processos de decisão de Markov no tempo contínuo. Contudo a função custo utilizada no problema de otimização representa a energia que a bateria libera para o sistema. Assim esse problema não oferece uma solução ótima para a maximização da autonomia da bateria, mas sim para a minimização da energia que ela fornece ao sistema.

Panigrahi et al.(24) introduziram um modelo de bateria como um processo estocástico discreto no tempo em que o estado do processo representa o estado de carga da bateria. Torna-se evidente então a possibilidade de adicionar esse modelo à abordagem de GDE através de processos de Markov no tempo discreto de Paleologo et al.(4). Contudo Panigrahi et al.(24) não apresentam uma função custo associada ao modelo de bateria, o que impede a sua utilização no problema de otimização da abordagem de GDE através de processos de Markov no tempo discreto de Paleologo et al.(4).

Rakhmatov e Vrudhula(12) desenvolveram um modelo analítico de bateria que possui uma função custo de bateria  $\sigma(T)$  (Equação (3.16)) que deve ser minimizada para que se maximize a autonomia da bateria. Na próxima seção é evidenciado como a função  $\sigma(T)$  pode ser aplicada ao problema de otimização da abordagem de GDE através de processos de Markov no tempo discreto de Paleologo et al.(4) e porque não é possível resolver esse

problema com essa função.

## 4.1 Função custo de bateria no problema de otimização de GDE

Seja um sistema modelado como processos de Markov no tempo discreto num estado  $x$  e é tomada uma decisão  $\delta_x$ . É definida a função  $c(s, a)$  como o consumo do PS quando no estado  $s$  e o comando  $a$  é escolhido. Tem-se a métrica de *nível esperado de consumo de potência*  $\bar{c}(x, \delta_x) = \sum_{p_a \in \delta_x} p_a c(s, a)$ . Um vetor de consumo é montado de acordo com a Equação (2.33). Paleologo et al.(4) enunciaram o problema de otimização PO2 (Equação (2.39)) em que é escolhida uma política  $\pi$  que minimiza o consumo do sistema  $\sum_{n=1}^{\infty} E [\bar{c}_{\delta(n)}^{\pi}]$  sob uma restrição de desempenho  $\sum_{n=1}^{\infty} E [d_{\delta(n)}^{\pi}] \leq D$ .

Como demonstrado por Rakhmatov e Vrudhula(12), a função de custo de bateria  $\sigma(T)$  (Equação (3.16)) deve ser minimizada para que se maximize a autonomia da bateria. Para utilizar  $\sigma(T)$  como função custo para o problema de otimização PO2 (Equação (2.39)), a função custo deve se reescrita como (4.1) em que a função  $F$  é definida na Equação (3.14). A métrica de *nível esperado de consumo* é expressa em (4.2), e o vetor de consumo é montado de acordo com a Equação (4.3). O consumo do sistema é então expresso como  $\sum_{n=1}^{\infty} E [\bar{b}_{\delta(n)}^{\pi}]$  e o problema de otimização da autonomia da bateria  $PObat$  é enunciado em (4.4).

$$b(s, a, t_k) = I(a, s, t_k)F(T, t_k, t_k + \Delta_k, \beta) \quad (4.1)$$

$$\bar{b}(x, \delta_x, t_k) = \sum_{p_a \in \delta_x} p_a b(s, a, t_k) \quad (4.2)$$

$$\bar{b}_{\delta(k)} := \begin{pmatrix} \bar{b}(x_1, \delta_{x_1}, t_k) \\ \vdots \\ \bar{b}(x_X, \delta_{x_X}, t_k) \end{pmatrix} \quad (4.3)$$

$$PObat : \min_{\pi} \sum_{n=1}^{\infty} E [\bar{b}_{\delta(n)}^{\pi}] \text{ tal que } \sum_{n=1}^{\infty} E [d_{\delta(n)}^{\pi}] \leq D \quad (4.4)$$

É importante perceber que a função custo  $b(s, a, t_k)$  usada em  $PObat$  contém uma forte dependência temporal que se origina da função  $F(T, t_k, t_k + \Delta_k, \beta)$ , à qual reproduz os efeitos não-lineares da bateria. Já a função custo  $c(s, a)$  de PO2 depende apenas do estado  $s$  do sistema e do comando  $a$ , i.e,  $c(s, a)$  será a mesma independente do instante de tempo  $t_k$ , o que permite simplificar PO2 para o programa linear PL1 (Equação (2.40)) em que não há a dependência temporal. Verifica-se então que a abordagem de Paleologo

et al.(4) não dá suporte à resolução do problema de otimização com uma função custo com dependência temporal, como é o problema *PObat*. Na Seção 4.2 é apresentada uma alternativa para o aumento da autonomia da bateria utilizando a abordagem de Chung et al.(1).

## 4.2 GDE orientado à autonomia da bateria em ambiente não-estacionário

Quando o sistema entra num estado de baixa energia que requer um menor valor de corrente da bateria, o efeito de relaxação faz a bateria “recuperar” carga disponível e então a autonomia pode ser consideravelmente estendida. Nessa Seção, é estendida a técnica de GDE proposta por Chung et al.(1), diminuindo o desempenho do sistema quando ele está em estados de baixa energia, o que aumenta o intervalo de tempo em que a bateria “recupera” e, conseqüentemente, a sua autonomia. A técnica proposta, denominada gerenciamento dinâmico de energia orientado à autonomia da bateria em ambiente não-estacionário (GDEBAT), é descrita nessa Seção para um PS de dois estados, *ligado* e *desligado*, mas ela pode ser estendida para lidar com um PS com mais estados combinando estados de baixa energia que requerem correntes nulas ou menores que a do estado de maior consumo de corrente, para que seja diminuído o desempenho do sistema sempre que o PS entre em um desses estados de baixa energia. Como apresentado no exemplo 12, quando ocorre uma redução de corrente, mesmo que essa não seja nula, ocorre o efeito de recuperação.

Quando baixo desempenho é aceitável, e o PS está desligado, ele pode levar certo tempo para religar, sob o custo de mais requisições pendentes esperando na FS. Seja  $T$  a resolução no tempo, i.e., o período de tempo em que o GE atua no PS. No caso do modelo em cadeia de Markov no tempo discreto do PS, o tempo de permanência esperado do estado *desligado* para o estado *ligado* ( $\bar{t}_{desligado,ligado}(ligar)$ ) quando o GE passa um comando *ligar* é inversamente proporcional à probabilidade de o PS transitar do estado *desligado* para o estado *ligado* ( $p_{desligado,ligado}^{PS}(ligar)$ ), como mostrado em (4.5). E, em geral, para estados do sistema em que o PS está desligado, a probabilidade de que o GE passe um comando *ligar* para o PS é menor quando a restrição de penalidade de desempenho  $D$  no problema de otimização apresentado por (4) é maior.

$$\bar{t}_{desligado,ligado}(ligar) = \frac{T}{p_{desligado,ligado}^{PS}(ligar)} \quad (4.5)$$

**Exemplo 14** Considere o sistema do exemplo 6. Resolvendo o problema de otimização para  $D = 0,3, 0,5, e 0,7$  são obtidas as tabelas de decisão mostradas em (4.6), (4.7) e

(4.8) respectivamente. Observa-se que as probabilidades para os estados (desligado,  $r_0, q_0$ ), (desligado,  $r_1, q_0$ ) e (desligado,  $r_1, q_1$ ) são as mesmas independentemente do  $D$  usado, mas para o estado (desligado,  $r_0, q_1$ ) a probabilidade de o GE passar um comando "ligar" diminui com o aumento do  $D$ .

$$M_\pi = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} (\text{ligado}, r_0, q_0) \\ (\text{ligado}, r_0, q_1) \\ (\text{ligado}, r_1, q_0) \\ (\text{ligado}, r_1, q_1) \\ (\text{desligado}, r_0, q_0) \\ (\text{desligado}, r_0, q_1) \\ (\text{desligado}, r_1, q_0) \\ (\text{desligado}, r_1, q_1) \end{matrix} & \begin{pmatrix} 0,0000 & 1,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 0,0000 & 1,0000 \\ 0,3299 & 0,6701 \\ 0,0000 & 1,0000 \\ 0,0000 & 1,0000 \end{pmatrix} \end{matrix} \quad (4.6)$$

$$M_\pi = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} (\text{ligado}, r_0, q_0) \\ (\text{ligado}, r_0, q_1) \\ (\text{ligado}, r_1, q_0) \\ (\text{ligado}, r_1, q_1) \\ (\text{desligado}, r_0, q_0) \\ (\text{desligado}, r_0, q_1) \\ (\text{desligado}, r_1, q_0) \\ (\text{desligado}, r_1, q_1) \end{matrix} & \begin{pmatrix} 0,0000 & 1,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 0,0000 & 1,0000 \\ 0,1086 & 0,8914 \\ 0,0000 & 1,0000 \\ 0,0000 & 1,0000 \end{pmatrix} \end{matrix} \quad (4.7)$$

$$M_\pi = \begin{matrix} & \begin{matrix} \text{ligar} & \text{desligar} \end{matrix} \\ \begin{matrix} (\text{ligado}, r_0, q_0) \\ (\text{ligado}, r_0, q_1) \\ (\text{ligado}, r_1, q_0) \\ (\text{ligado}, r_1, q_1) \\ (\text{desligado}, r_0, q_0) \\ (\text{desligado}, r_0, q_1) \\ (\text{desligado}, r_1, q_0) \\ (\text{desligado}, r_1, q_1) \end{matrix} & \begin{pmatrix} 0,0000 & 1,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 1,0000 & 0,0000 \\ 0,0000 & 1,0000 \\ 0,0615 & 0,9385 \\ 0,0000 & 1,0000 \\ 0,0000 & 1,0000 \end{pmatrix} \end{matrix} \quad (4.8)$$

Se um sistema alimentado a bateria passa mais tempo em estados de baixo consumo de energia, ele se beneficia do efeito de relaxação e estende a autonomia da bateria. A técnica GDE aqui apresentada vai usar: políticas de energia otimizadas para restrições baixas de penalidade de desempenho em estados de alta energia; e políticas de energia otimizadas para restrições altas de penalidade de desempenho em estados de baixa energia.

Num dado intervalo de tempo, uma estimativa  $(\hat{R}_0, \hat{R}_1)$  é obtida (conforme os métodos apresentados na Seção 2.2) e são escolhidos dois índices consecutivos para cada dimensão:  $R_{0j} \leq \hat{R}_0 \leq R_{0(j+1)}$  e  $R_{1k} \leq \hat{R}_1 \leq R_{1(k+1)}$ . O cálculo da tabela de decisão para o estado do sistema e o  $(\hat{R}_0, \hat{R}_1)$  utiliza quatro tabelas de decisão correspondentes a esses índices escolhidos e a penalidade de desempenho  $D$  a ser utilizada, havendo em cada uma dessas tabelas uma linha associada ao estado atual do sistema, denotada por  $EA$ . Uma técnica de interpolação bi-dimensional (Seção 2.2) é aplicada a essas quatro linhas para calcular-se a linha de decisão final. Assim, a escolha de cada uma das quatro tabelas de decisão utilizadas para obter a decisão final é baseada em três parâmetros: a probabilidade  $R0$  de o RS continuar no estado em que não emite requisições, a probabilidade  $R1$  de o RS continuar no estado em que emite requisições, e a penalidade de desempenho  $D$  a ser usada, como ilustrado na Figura 4.1. As tabelas de decisão são armazenadas num *espaço de políticas*.

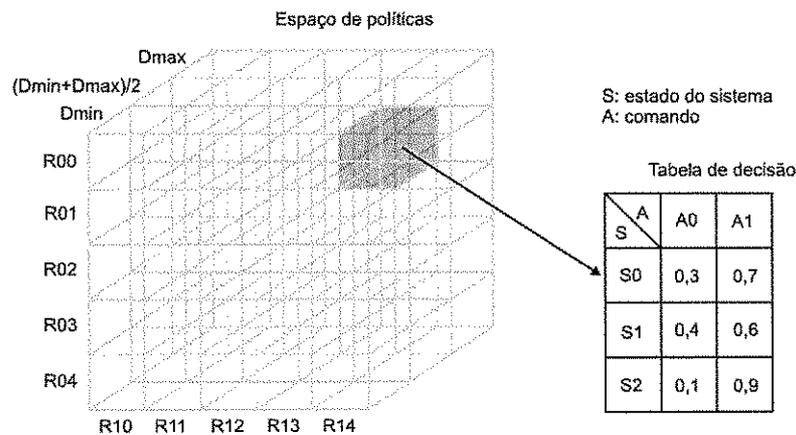


Figura 4.1: Escolha de tabela de decisão com base na identificação do RS e na penalidade de desempenho a ser usada.

Considere o PS do exemplo 5. Quando o PS está ligado, uma restrição mínima de penalidade de desempenho  $D_{min} = 0,2$  é desejada. E quando o PS está desligado, uma restrição máxima de penalidade de desempenho  $D_{max} = 0,4$  é aceitável. Imediatamente após o PS desligar, não é desejado que o GE use uma política de energia otimizada para a alta restrição de penalidade de desempenho  $D_{max} = 0,4$ , pois, se assim, o usuário do sistema vai experimentar um atraso elevado para religar o sistema. Então a transição entre  $D_{min} = 0,2$  para  $D_{max} = 0,4$  deve ser realizada suavemente.

É usada uma função exponencial para associar intervalos de valores que essa função pode assumir a valores de  $D$  entre 0,2 e 0,4 como mostrado na Figura 4.2. A função exponencial  $f(t_{desligado})$  é mostrada na Equação (4.9), onde  $t_{desligado}$  é o tempo que o PS permanece no estado *desligado*,  $\bar{t}_r(\epsilon)$  é o limitante superior para o tempo de recuperação

(Equação (3.22)) da bateria usada.

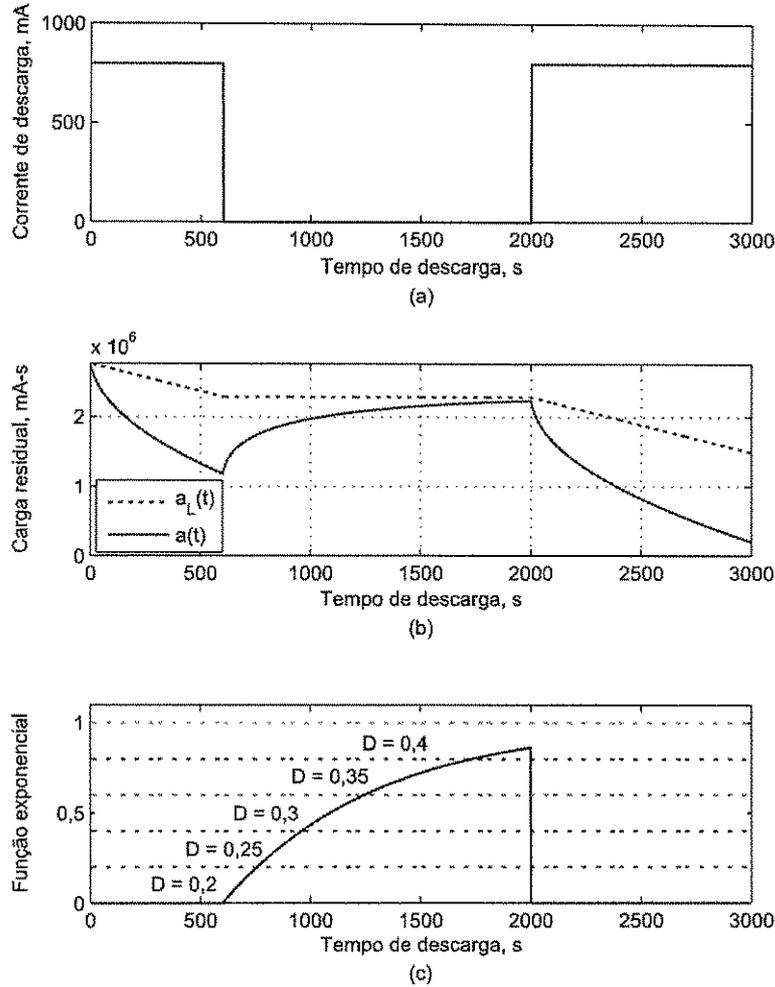


Figura 4.2: Exemplo de uso de uma função exponencial para variar restrições de penalidade de desempenho quando o PS desliga.

$$f(t_{\text{desligado}}) = 1 - e^{-t_{\text{desligado}}/\bar{t}_r(\varepsilon)} \quad (4.9)$$

Seja  $N_D$  o número de valores de penalidade de desempenho igualmente espaçados  $D(i_D)$  (veja Equação (4.10)), onde  $i_D = 0, 1, 2, \dots, N_D$ . É escolhida o valor de restrição de penalidade de desempenho  $D(i_D)$  tal que  $(i_D - 1)/N_D \leq f(t_{\text{desligado}}) \leq i_D/N_D$ .

$$D(i_D) = \frac{(i_D - 1)(D_{\max} - D_{\min})}{N_D - 1} + D_{\min} \quad (4.10)$$

**Exemplo 15** Seja a FS do exemplo 2, o RS não-estacionário especificado em (2.52) e (2.53), e o PS modelado pelo consumo de corrente (em mA) especificado na Equação (4.11), a taxa de serviço apresentada em (4.12) e a matriz de probabilidades de transições

em (4.13) e (4.14). Deseja-se simular esse sistema durante 10000 passos de simulação, o que resulta num fator de desconto  $\xi = 1 - 1/10000 = 0,9999$ . Para identificar o modelo do RS, foi utilizada a abordagem de multi-janela (1) com janela de comprimento 100.

$$c(s, a) = \begin{array}{cc} & \begin{array}{cc} \text{ligar} & \text{desligar} \end{array} \\ \begin{array}{c} \text{ligado} \\ \text{desligado} \end{array} & \begin{bmatrix} 200 & 250 \\ 250 & 0 \end{bmatrix} \end{array} \quad (4.11)$$

$$b(s, a) = \begin{array}{cc} & \begin{array}{cc} \text{ligar} & \text{desligar} \end{array} \\ \begin{array}{c} \text{ligado} \\ \text{desligado} \end{array} & \begin{bmatrix} 0,8 & 0 \\ 0 & 0 \end{bmatrix} \end{array} \quad (4.12)$$

$$P^{PS}(\text{ligar}) = \begin{array}{cc} & \begin{array}{cc} \text{ligado} & \text{desligado} \end{array} \\ \begin{array}{c} \text{ligado} \\ \text{desligado} \end{array} & \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \end{array} \quad (4.13)$$

$$P^{PS}(\text{desligar}) = \begin{array}{cc} & \begin{array}{cc} \text{ligado} & \text{desligado} \end{array} \\ \begin{array}{c} \text{ligado} \\ \text{desligado} \end{array} & \begin{bmatrix} 0,8 & 0,2 \\ 0 & 1 \end{bmatrix} \end{array} \quad (4.14)$$

Para GDEBAT, cinco valores de restrições de penalidade de desempenho foram usadas:  $D = 0,5, 0,525, 0,55, 0,575$  e  $0,6$ . Foi usado o modelo de bateria de Rakhmatov e Vrudhula(12) com parâmetros  $\alpha = 2838374,0484 \text{mA} \cdot \text{s}$  e  $\beta = 0,0364 \text{s}^{-\frac{1}{2}}$ . O limitante superior para o tempo de recuperação (Equação (3.22)) foi calculado para  $\varepsilon = 0,39$ . Simulou-se também o sistema com a técnica de GDE de Chung et al.(1) com  $D = 0,5$  para fins de comparação. A restrição de perda de requisições foi fixada em 15% para as duas técnicas. Quatro métricas foram calculadas para avaliar a qualidade dos algoritmos:

- $I$ : Corrente média (unidade: mA)
- $\hat{D}$ : Número médio de requisições pendentes na FS por período (%)
- $\hat{L}$ : Perda média de requisições por período (%)
- $B$ : Autonomia da bateria (unidade: s)

Os resultados de simulação são apresentados na Tabela 4.1. A autonomia da bateria do sistema com a técnica GDEBAT foi 81,95% maior do que com o algoritmo de Chung et al.(1) à custa de um aumento em 1,29% na perda média de requisições, que continuou respeitando a restrição de perda de requisições.

Tabela 4.1: Comparação de algoritmos.

Algoritmo	I (mA)	$\hat{D}$	$\hat{L}$	B (s)
Chung et al.(1) ( $D = 0,5$ )	362,73	30,74%	11,88%	5253
GDEBAT ( $0,5 \leq D \leq 0,6$ )	219,33	53,37%	13,17%	9558

### 4.3 Conclusões

A solução proposta por Chung et al.(1) otimiza o consumo de energia, que seria a solução ótima caso a bateria fosse uma fonte ideal. Contudo baterias reais apresentam a característica de taxa de capacidade e o efeito de recuperação, que não são contemplados em modelos lineares de bateria. Então a técnica de Chung et al.(1) pode vir a não otimizar a autonomia de dispositivos reais alimentados a bateria. Para superar essa limitação, GDEBAT reduz o desempenho do sistema a níveis aceitáveis quando a carga de trabalho é baixa. O sistema permanece mais tempo em estados de baixo consumo de energia, tirando proveito do efeito de recuperação e assim aumentando a autonomia da bateria.

No Capítulo 5 é mostrado como GDEBAT pode ser implementada num sistema embarcado. É utilizada a plataforma OMAP1611 e apresentada a metodologia necessária para utilização dos recursos de gerenciamento de energia da plataforma alvo, caracterização da bateria, aquisição de dados de corrente e tensão que a bateria fornece ao sistema, e modelagem do sistema. A discussão sobre a avaliação do desempenho de GDEBAT em relação a outras escolhas de gerenciamento de energia é abordada através de um estudo de caso.

## Capítulo 5

# Metodologia experimental

Neste capítulo são apresentados os recursos disponíveis para realização de gerenciamento de energia na plataforma alvo em que serão realizadas a implementação e avaliação da política de gerenciamento de energia proposta por esse trabalho, além da metodologia de medição de tensão e corrente que a bateria fornece à plataforma. A plataforma alvo é o OMAP1611 (Plataforma Aberta para Aplicações Multimídia) da Texas Instruments. O OMAP1611 é uma plataforma com dois núcleos: um processador digital de sinais (PDS) TMS320C55x e um processador ARM926TEJ. O DSP é apropriado para operações de computação intensiva, e.g. multimídia, enquanto o sistema operacional e a interface com o usuário são executados no ARM (26).

### 5.1 Gerenciamento de energia no OMAP1611 através do projeto DPM

O projeto de código aberto *Dynamic Power Management* (DPM) (27) provém da colaboração entre *IBM Austin Research Lab* e *MontaVista Software*. Seu objetivo é melhorar a conservação de energia de plataformas diversas tais como: Intel Centrino, OMAP e IBM PowerPC 405LP. E dentre as plataformas OMAP's atualmente suportadas pelo projeto DPM estão: OMAP730, OMAP1510, OMAP5910 e OMAP1610. Devido ao suporte ao OMAP1610, o projeto DPM também pode ser usado no OMAP1611.

O projeto DPM é um módulo do sistema operacional que exerce gerenciamento ativo de energia (28). Tanto a partir do nível de aplicação como do núcleo do sistema operacional, os gerenciadores de políticas de DPM e aplicações podem usar esse módulo através de uma *Application Programming Interface* (API). A partir dessas considerações, ressalta-se que o projeto DPM não é classificado como um algoritmo de escalonamento dinâmico de tensão e frequência (EDTF), ou um mecanismo de controle de gerenciamento de energia tal como a Interface Avançada de Configuração e Energia (ACPI), ou como um sistema

operacional consciente em relação ao consumo de energia (*power-aware*).

### 5.1.1 Políticas DPM

Alguns conceitos fundamentais no projeto DPM (28):

**Ponto de operação** reúne um conjunto mínimo de parâmetros físicos discretos e interdependentes que caracterizam um nível específico de desempenho do sistema, que acarreta num determinado consumo de energia. Um ponto de operação pode ser definido por parâmetros como tensão e frequência de operação da UCP, frequências de barramentos, etc.

**Estado de operação** representa o estado de atividade/inatividade do sistema. Por exemplo, na Figura 5.1, à medida que ocorrem eventos, um sistema transita entre estados: tarefas são agendadas, interrupções são gerenciadas, o sistema fica desocupado, o sistema entra no modo de dormir (*sleep*) e sai dele. Cada um desses estados é considerado um estado de operação.

**Política DPM** é uma estrutura de dados que mapeia estados de operação a pontos de operação.

**Gerenciador de política** é um executável que controla a ativação de políticas.

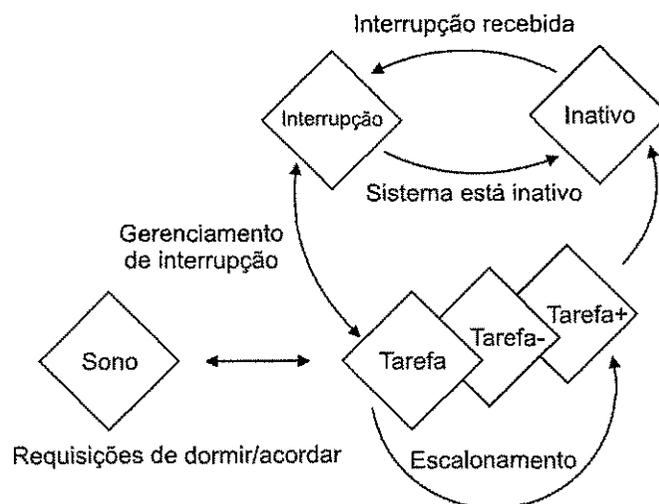


Figura 5.1: Exemplo de estados de operação e transições candidatas a uma implementação no projeto DPM.

Num sistema com o projeto DPM, os componentes interagem como ilustrado na Figura 5.2.

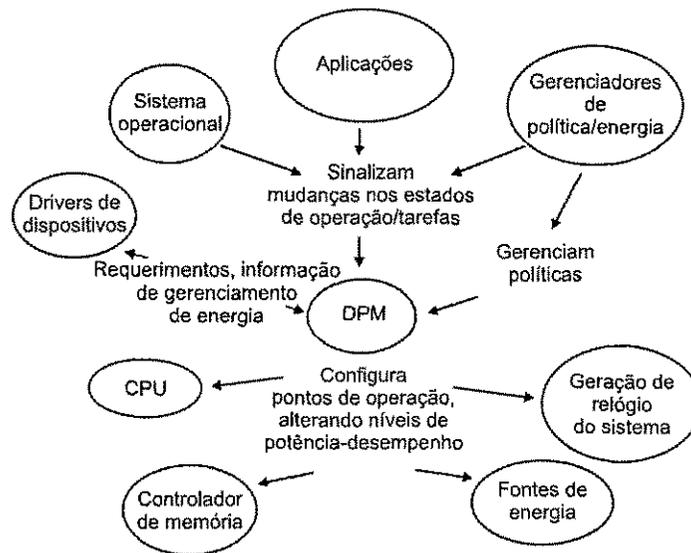


Figura 5.2: Interações entre componentes de um sistema com DPM.

Na plataforma OMAP1611, os pontos de operação definidos pelos utilitários padrão do projeto DPM (6) são mostrados na Tabela 5.1. Nessa Tabela, os pontos de operação são especificados pelas frequências de operação do *digital phase-locked loop* (DPLL), da *Unidade Central de Processamento* (UCP) e do *Traffic Controller* (TC).

Tabela 5.1: Alguns pontos de operação do OMAP1611

Nome	Descrição	Observação
192	UCP a 192 MHz	
168	DPLL e UCP a 168 MHz, TC a 84 MHz	
84	DPLL a 168 MHz, UCP e TC a 84 MHz	
60	DPLL, UCP e TC a 60 MHz	
sono-168	Modo Sono, DPLL a 168 MHz	Modo Sono Profundo
sono-60	Modo Sono, DPLL a 6 MHz, TC a 3MHz	Modo Sono Profundo

Na Tabela 5.2 são mostradas duas políticas de energia, *alta-potência* e *baixa-potência* (29). Os estados de operação são mapeados a pontos de operação presentes na Tabela 5.1.

A política DPM usada no OMAP pode ser trocada em tempo de execução. A Figura 5.3 mostra a tensão, corrente e potência de saída de uma bateria do tipo Li-Ion de 3,7V conectada a um OMAP 1611 inicialmente com a política “alta-potência” que depois é trocada pela política “baixa-potência”. Depois de 1 segundo, é ativada a política “alta-potência” que permanece durante mais 1 segundo e esse ciclo é repetido mais uma vez. O tempo de transição entre as políticas “alta-potência” e “baixa-potência” é de 6ms.

Estado de operação	alta-potência	baixa-potência
tarefa-inativa	192	60
inativo	192	60
sono	sono-168	sono-60
tarefa-4	168	60
tarefa-3	168	60
tarefa-2	168	60
tarefa-1	168	60
tarefa	192	60
tarefa+1	192	84
tarefa+2	192	84
tarefa+3	192	84
tarefa+4	192	84

Tabela 5.2: Exemplos de mapeamentos de pontos de operação do OMAP1510 a estados de operação

Evidencia-se que, devido ao fato de haver redução de corrente da bateria na transição entre as políticas “alta-potência” e “baixa-potência”, mesmo sendo a corrente não nula nessa segunda política, ocorre o efeito de recuperação da bateria no período de tempo em que o OMAP permanece na política “baixa-potência”.

### 5.1.2 Uso de temporizadores de propósito geral no OMAP1611

Os temporizadores de propósito geral (ou de Modo Dual) podem ser usados para modulação por largura de pulso programável e religamento programado, pois podem continuar ligados enquanto a plataforma está no estado de Sono Profundo (30). Como os temporizadores de propósito geral estão num barramento compartilhado, tanto o ARM quanto o DSP podem utilizá-los. Temporizadores de propósito geral podem ser usados para religar o OMAP após um período especificado de tempo em que ele fica em Sono Profundo. O projeto DPM é ativado num OMAP 161x com a política **alta-potência** mostrada na Tabela 5.2, onde:

- O estado padrão do sistema quando ligado é “tarefa”, com o ponto de operação “192”;
- O estado padrão do sistema quando em sono profundo é “sono”, com o ponto de operação “sono-168”;

Um temporizador é configurado para acordar o sistema 5 segundos após ele entrar em Sono Profundo. Na Figura 5.4 são mostradas a tensão, corrente e potência de saída de

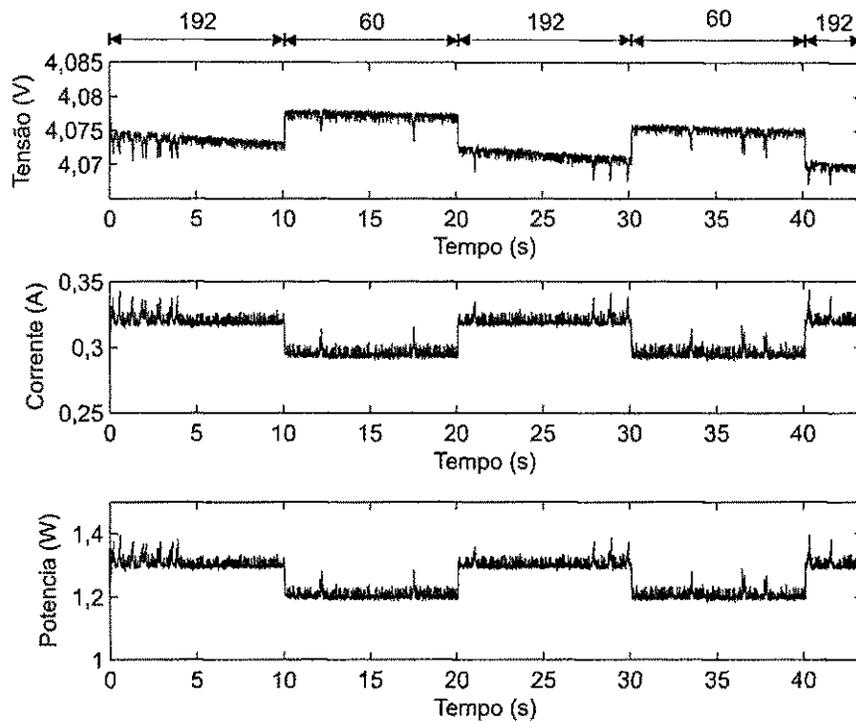


Figura 5.3: Transição entre políticas de energia “alta-potência” e “baixa-potência”, cujos pontos de operação no estado “tarefa” são “192” e “60” respectivamente.

uma bateria do tipo Li-Ion de 3,7V conectada ao OMAP 1611 quando é colocado em Sono Profundo (ponto de operação sono-168) durante 5 segundos. Evidencia-se que, devido ao fato de haver redução de corrente da bateria entre os estados “tarefa” e “sono”, mesmo sendo a corrente não nula nesse segundo estado, ocorre o efeito de recuperação da bateria durante o Sono Profundo.

O projeto DPM pode ser configurado num OMAP 1611 com a política **baixa-potência** mostrada na Tabela 5.2, onde:

- O estado padrão do sistema quando ligado é “tarefa”, com o ponto de operação “60”;
- O estado padrão do sistema quando em sono profundo é “sono”, com o ponto de operação “sono-60”;

Na Figura 5.5 são mostradas a tensão, corrente e potência de saída de uma bateria do tipo Li-Ion de 3,7V conectada ao OMAP 1611 quando é colocado em Sono Profundo (ponto de operação sono-60) durante 5 segundos.

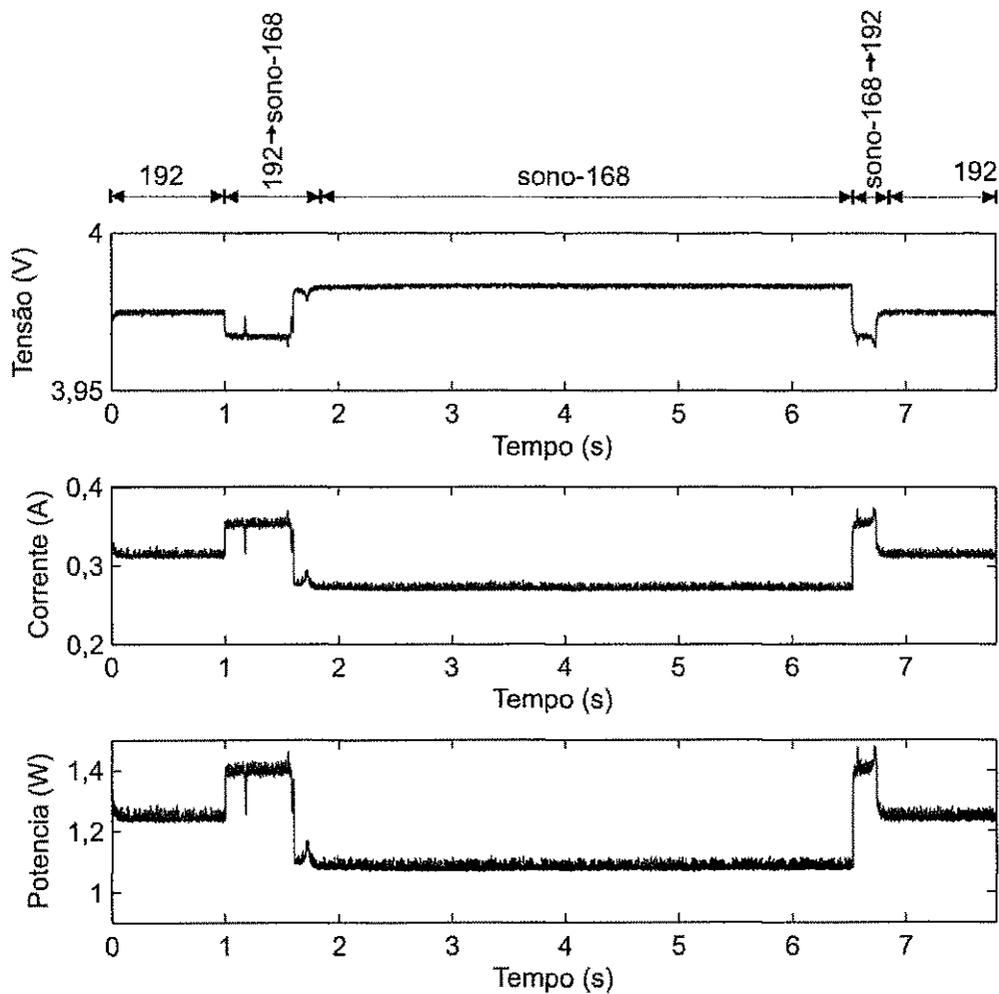


Figura 5.4: Transição entre pontos de operação 192 e sono-168, estado em que permanece o OMAP por 5 segundos e depois retorna ao estado 192.

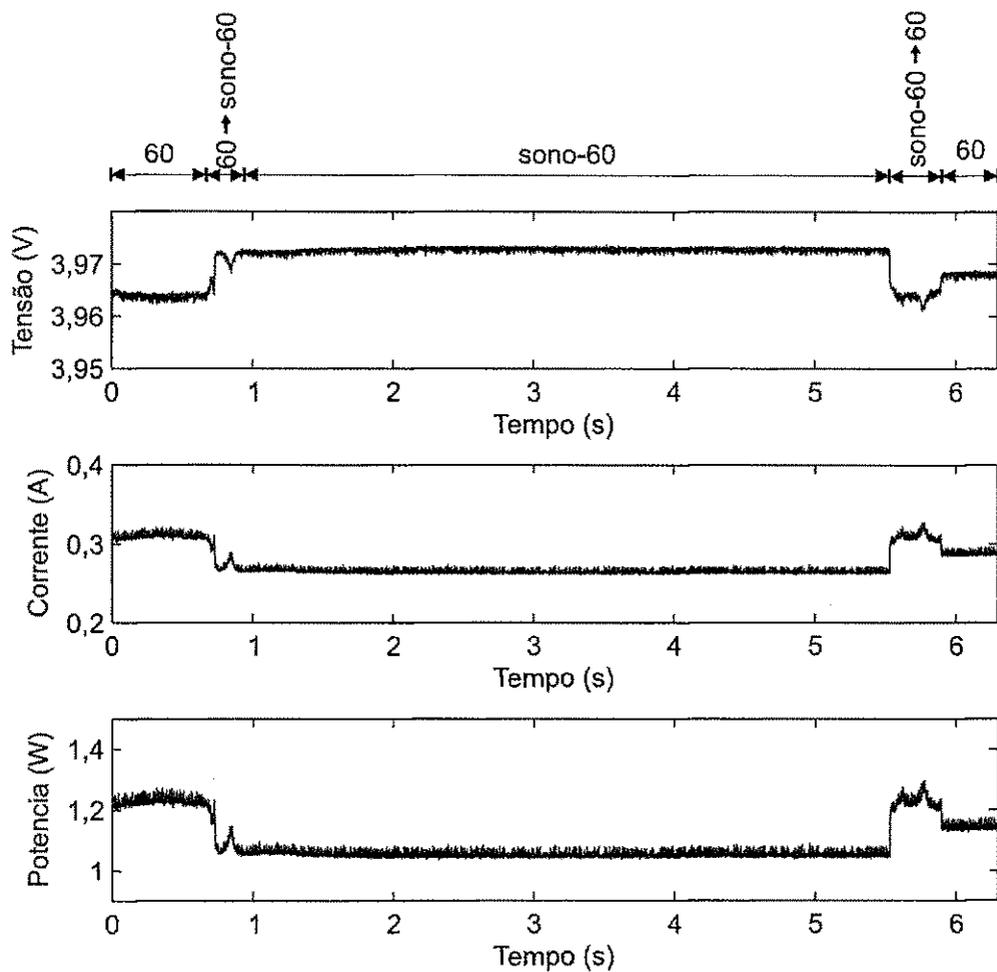


Figura 5.5: Transição entre pontos de operação 60 e sono-60, estado em que permanece o OMAP por 5 segundos e depois retorna ao ponto de operação sono-60.

## 5.2 Medição da tensão e corrente da bateria

Para medir a tensão entre os terminais da bateria e a corrente que ela fornece ao OMAP, foi utilizado o módulo de aquisição de dados NI USB-6210 conectado a um *computador pessoal* (PC) via *Universal Serial Bus* (USB), como ilustrado na Figura 5.6. No PC foi utilizado o programa LabVIEW no sistema operacional Windows

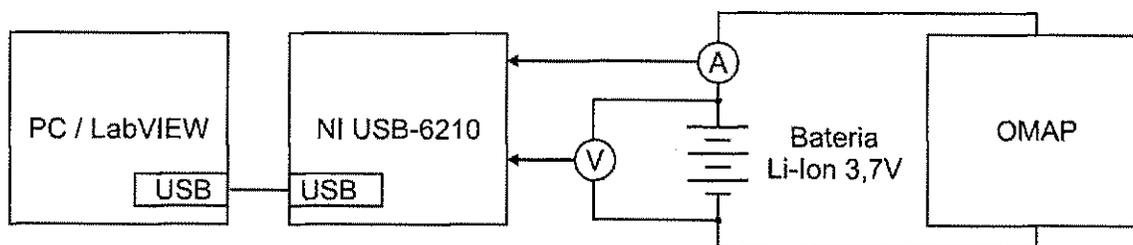


Figura 5.6: Diagrama de blocos para medição da tensão e corrente da bateria.

Conforme a Figura 5.7, para medir o valor  $v$  da tensão entre os terminais positivo  $VBAT+$  e negativo  $VBAT-$  da bateria, o canal de entrada analógico A/2 do módulo de aquisição de dados NI USB-6210 foi conectado ao terminal  $VBAT+$  da bateria, e o canal A/GND do NI USB-6210 foi conectado ao terminal  $VBAT-$  da bateria. O terminal  $THERM$  da bateria permite a medição da temperatura da bateria (que pode ser importante, pois a temperatura influencia no comportamento não linear da bateria, mas o presente trabalho assume, como hipótese simplificadora, que a temperatura da bateria é constante).

Para medir o valor  $i$  da corrente da bateria que flui do terminal negativo  $VBAT-$  do OMAP para o terminal  $VBAT-$  da bateria, foi colocada uma resistência  $R$  entre esses dois terminais. O canal A/1 do NI USB-6210 foi conectado ao terminal  $VBAT-$  do OMAP. Assim o canal A/1 do NI USB-6210 mede a tensão  $Ri$  sobre a resistência  $R$ , que é proporcional à corrente  $i$ . No programa LabVIEW, o valor da resistência  $R$  é especificado, e os valores adquiridos no canal canal A/1 são automaticamente divididos por  $R$ , já fornecendo diretamente os valores de corrente  $i$ .

A tensão nominal da bateria é de  $3,7V$ . Durante o ligamento da placa no acionamento da sua chave, pode ser observada por um intervalo de tempo de  $1,1s$  uma corrente de  $584mA$ . Contudo ao longo do funcionamento normal do OMAP, a máxima corrente observada foi de  $380mA$ , na transição entre os pontos de operação sono-168 e 192. Assim, durante o funcionamento normal do OMAP, a queda de tensão máxima que uma resistência  $R = 0,2\Omega$  causará é de  $76mV$ , o que não interfere no funcionamento do OMAP. Foi utilizada então uma resistência  $R = 0,2\Omega$  de  $5W$  e tolerância de 5%.

O diagrama de blocos para aquisição de dados no programa LabVIEW é mostrado na Figura 5.8. O bloco *DAQ Assistant* é o driver do LabVIEW com o NI USB-6210 e sua

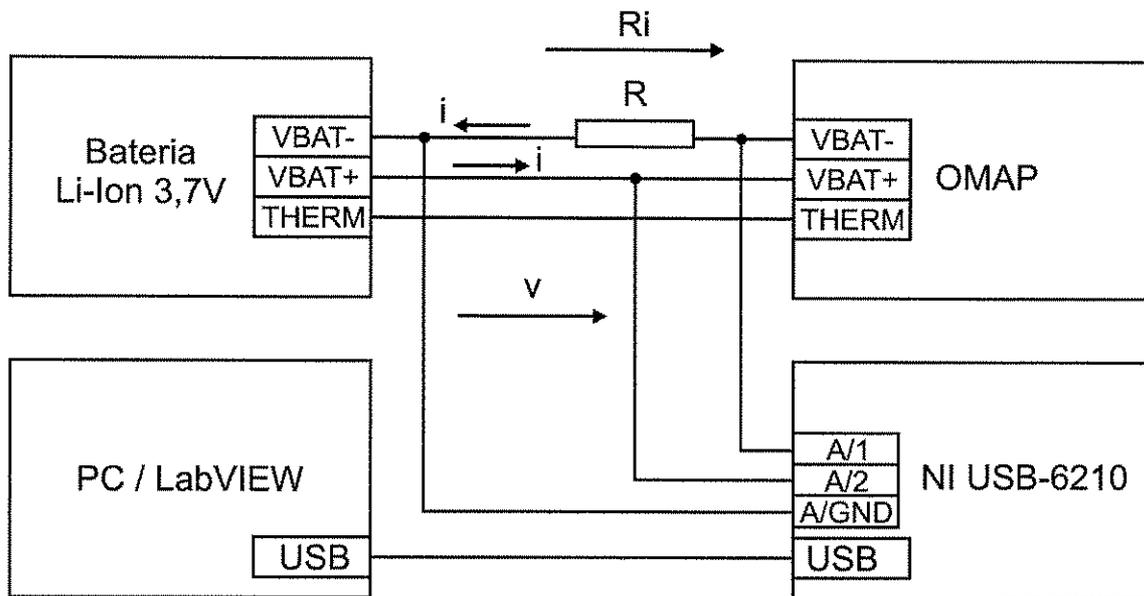


Figura 5.7: Diagrama elétrico para medição da tensão e corrente da bateria através do módulo de aquisição de dados NI USB-6210.

saída são os sinais de tensão e corrente. Esses sinais são multiplicados para obtenção da potência dissipada pelo OMAP e são exibidos no painel frontal do LabVIEW em valor instantâneo e como forma de onda, conforme a Figura 5.9. O bloco *Write LabVIEW Measurement File* salva em arquivo os valores de tensão, corrente e potência.

### 5.3 Caracterização da bateria para validação de GDE-BAT

Como descrito na Seção 3.1, a caracterização de uma bateria segundo o modelo de Rakhmatov e Vrudhula(12) exige a determinação experimental dos parâmetros  $\alpha$  e  $\beta$ , o que é realizado através de uma série de ensaios com corrente constante. Para isso foi utilizado o sistema de caracterização da bateria ilustrado na Figura 5.10. Uma fonte de corrente constante  $I_D$  drena corrente da bateria, que é medida pelo módulo de aquisição de dados NI USB-6210, assim como a tensão entre seus terminais. O diagrama elétrico do circuito utilizado como fonte de corrente é ilustrado na Figura 5.11. Invertendo-se o sentido da bateria, tal circuito pode ser utilizado tanto para o processo de descarregar como o de carga da bateria.

Para realização dos experimentos com a política de energia proposta, foi utilizada uma bateria Siemens Li-Ion modelo V30145-K1310-X329 de tensão nominal de 3,7V, tensão de corte de 3,37V, e carga nominal de 820mAh. A essa bateria foram aplicadas várias

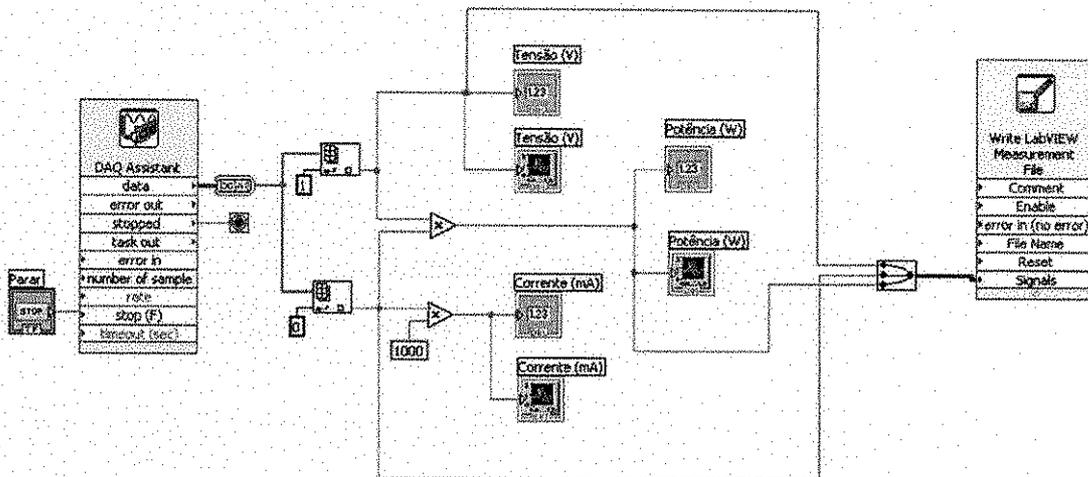


Figura 5.8: Diagrama de blocos no programa LabVIEW.

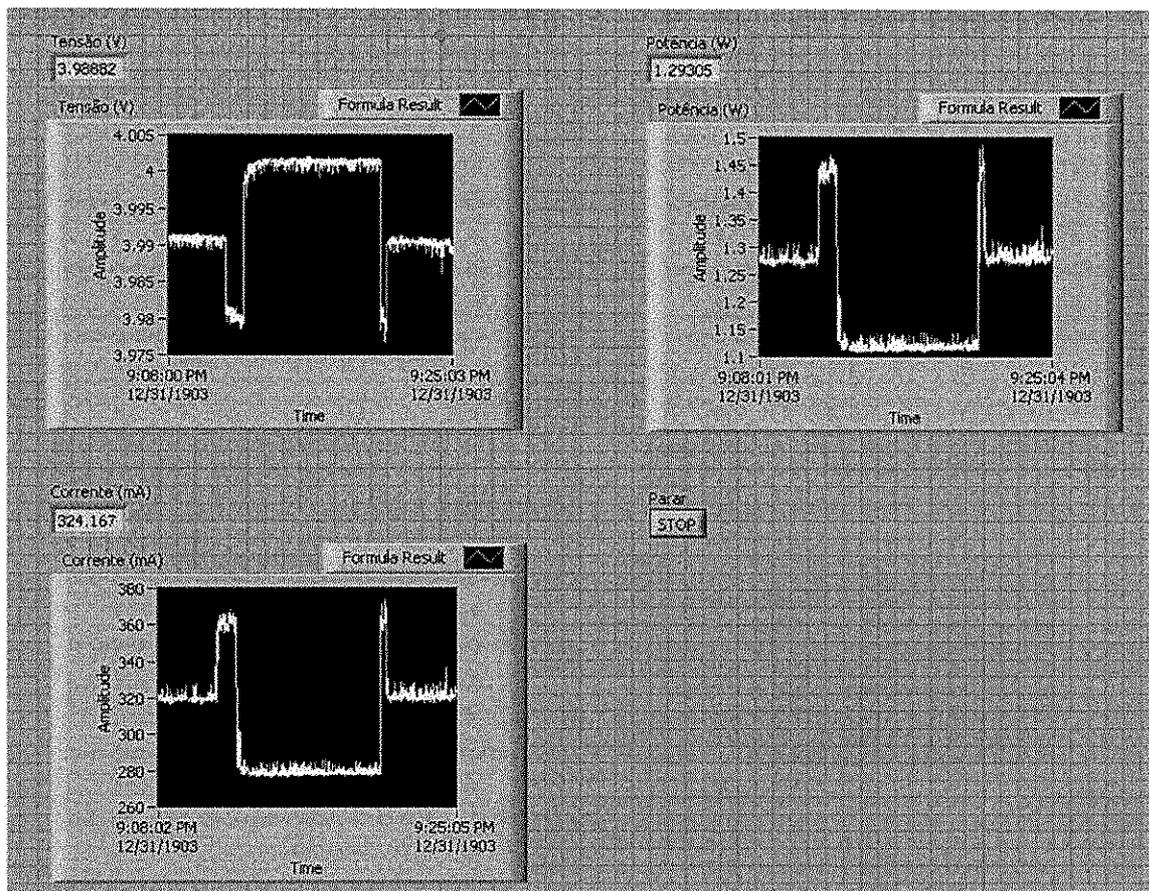


Figura 5.9: Painel frontal no programa LabVIEW.

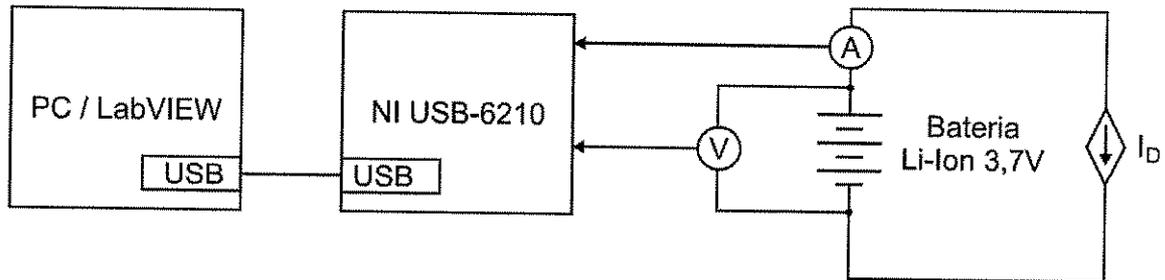


Figura 5.10: Esquema para ensaios a corrente constante para caracterização da bateria.

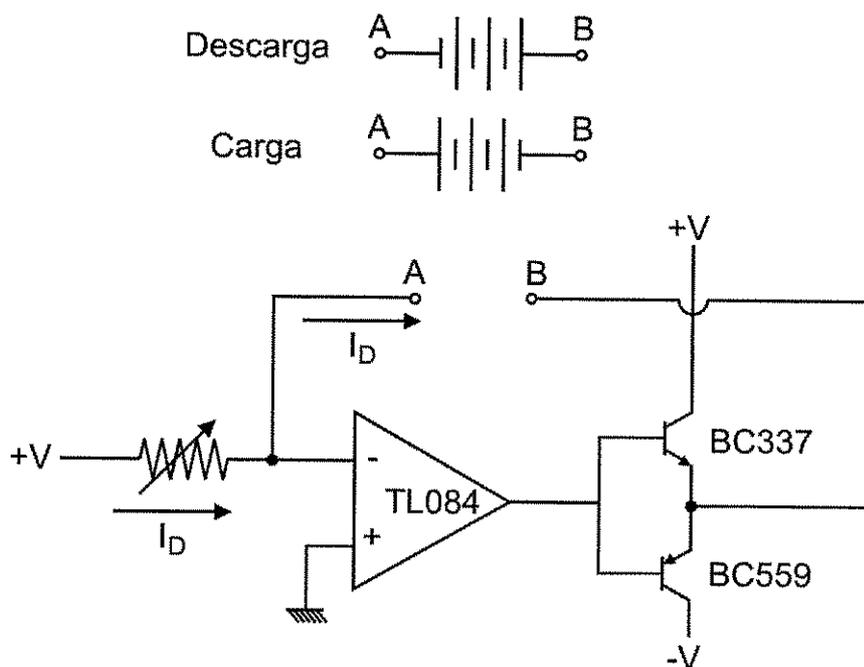


Figura 5.11: Diagrama elétrico do circuito utilizado para descarga ou carga da bateria a corrente constante.

correntes constantes e para cada uma foi obtido um tempo de falha (autonomia), conforme a Tabela 5.3. Os parâmetros da bateria são estimados segundo a técnica apresentada na Seção 3.1, resultando em  $\alpha = 2838374,0484mA \cdot s$  e  $\beta = 0,0364s^{-2}$ . Na Figura 5.12 é mostrada a curva de corrente de descarga versus autonomia para os dados experimentais da tabela 5.3 e a curva calculada com os valores estimados de  $\alpha$  e  $\beta$ .

$I_{(k)}$ (mA)	$L_{(k)}$ (min)
21,4	126180
26,4	104880
41,2	65700
48,9	55020
58,6	45540
69,6	40620
82,6	31260
95,3	27540
119,4	23220
255,1	10920
310,0	8580

Tabela 5.3: Autonomias de uma bateria Siemens Li-Ion de 3,7V e 820mAh para diferentes ensaios a corrente constante.

## 5.4 Implementação de GDEBAT

Para a implementação da política de gerenciamento de energia na plataforma alvo, o código em Matlab de simulação foi transcrito manualmente para a linguagem de programação C e compilado diretamente na plataforma alvo OMAP1611. A implementação da política de energia é composta de duas partes: cálculo do espaço de políticas antes do tempo de execução, e aplicação em tempo de execução, cujos diagramas de fluxo são apresentados nas Figuras 5.13 e 5.14 respectivamente.

O cálculo da tabela de políticas toma por entradas o modelo do sistema e as restrições de desempenho e resolve o problema de otimização (Seção 2.1) para cada:  $R0$  que varia com  $indiceRS1 = 1, \dots, NS1$ ;  $R1$  que varia com  $indiceRS2 = 1, \dots, NS2$ ;  $D$  que varia com  $indiceD = 1, \dots, ND$ . Cada política ótima obtida é armazenada num espaço de políticas que ao final é salvo num arquivo de cabeçalho .h o qual é incluído pela aplicação em tempo de execução.

A aplicação em tempo de execução inicialmente abre o arquivo provido pelo projeto DPM para ser possível realizar as trocas de estados de energia do OMAP1611. Segue-se

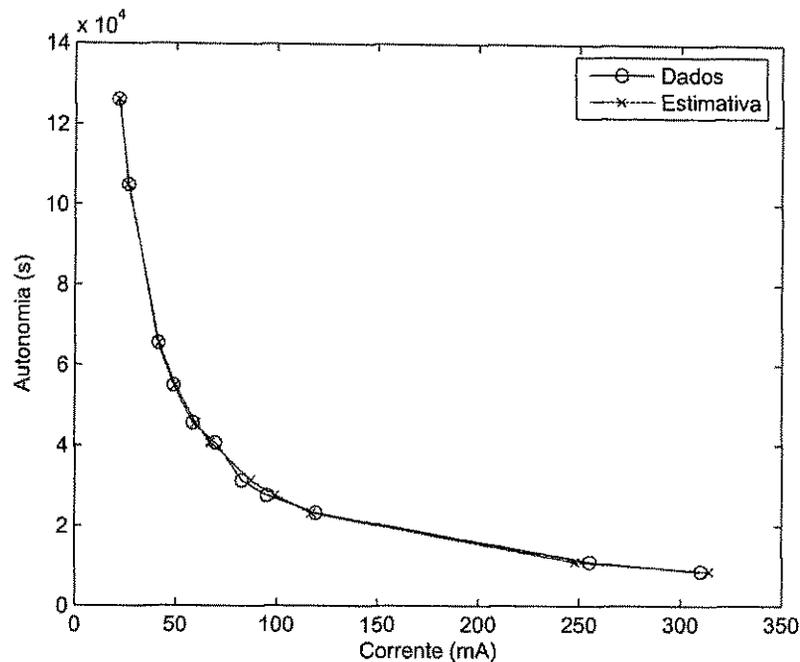


Figura 5.12: Ensaio a corrente constante para caracterização da bateria.

então um laço que é repetido  $N$  vezes, em que  $N$  é um número suficientemente grande para exaurir a bateria. Numa aplicação final a condição de parada desse laço seria um pedido do usuário para não mais utilizar a política. Nesse laço: são observados os estados do sistema; é identificado o RS (Seção 2.2.2); é escolhida a penalidade de desempenho a usar e calculada a decisão final (Seção 4.2); o GE escolhe um comando e o aplica ao PS escrevendo no arquivo para troca do estado de energia. Ao final da aplicação é fechado o arquivo provido pelo projeto DPM para trocas de estados de energia do OMAP1611

## 5.5 Estudo de caso

Para avaliar GDEBAT, foi implementado na plataforma OMAP 1611 o RS não estacionário em que as probabilidades de transição variam conforme (2.52) e (2.53), e a FS do exemplo 2. A política de gerenciamento de energia será executada a cada período de 1s.

O provedor de serviço é o OMAP, que nesse estudo de caso irá assumir dois possíveis estados, 192 e 60, presentes na Tabela 5.1. Através da metodologia apresentada na Seção 5.2, foi obtida a matriz  $c(s, a)$  da Equação (5.1), que especifica o consumo de corrente (em mA) do PS no estado  $s \in \{192, 60\}$ , e quando o comando  $a \in \{trocarPara192, trocarPara60\}$  é passado pelo GE. Os comandos *trocarPara192* e *trocarPara60* significam que o GE enviará um comando para o projeto DPM (Seção 5.1) configurar o OMAP para o ponto de operação 192 ou 60 respectivamente. O aumento de corrente para realização das trocas

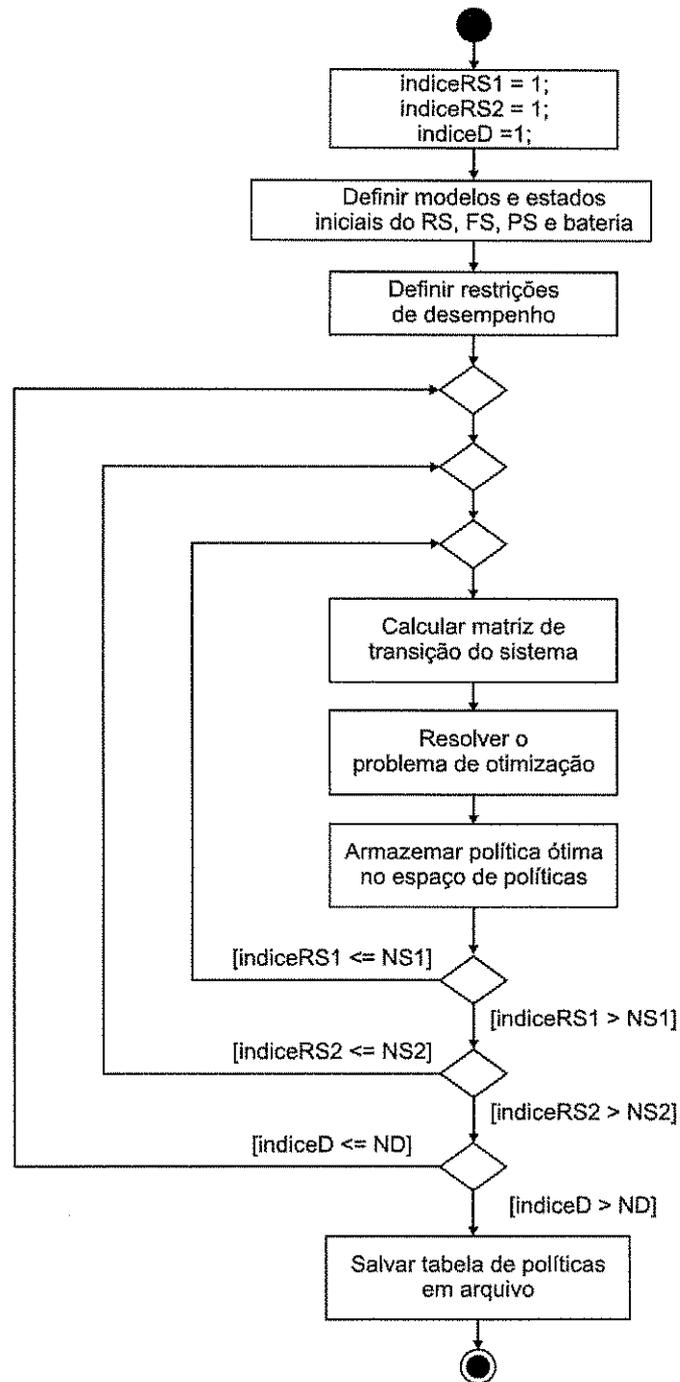


Figura 5.13: Diagrama de fluxo do cálculo da tabela de políticas antes do tempo de execução.

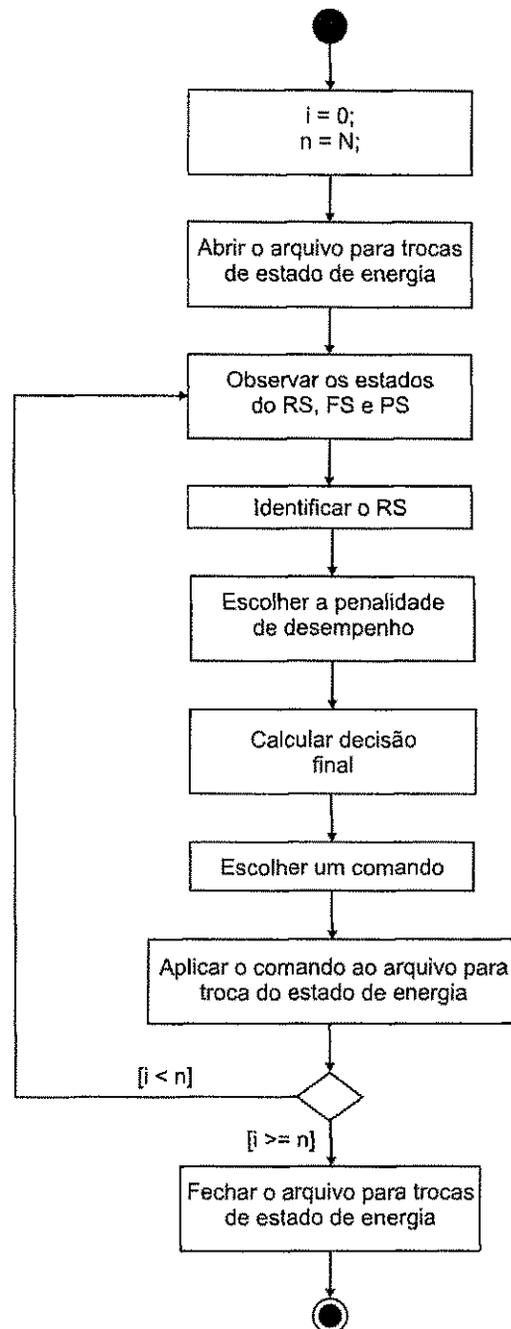


Figura 5.14: Diagrama de fluxo da política de gerenciamento de energia em tempo de execução.

de estado do PS é desprezível. Assim a corrente média para um dado estado do PS independe do comando do GE. A matriz  $b(s, a)$  da Equação (5.2) especifica a taxa de serviço do PS no estado  $s \in \{192, 60\}$  e quando o comando  $a \in \{trocarPara192, trocarPara60\}$  é passado pelo GE, i.e., a probabilidade de servir uma requisição num período. A taxa de serviço para o estado 192 é 1 pois a requisição possui tempo médio de atendimento de 1s. A taxa de serviço para o estado 60 é 0,3125 pois a diminuição de frequência de 192MHz para 60MHz faz com que uma requisição que seria atendida com tempo médio de 1s para o PS em 192MHz, é atendida com  $192/60 = 3,2s$  com o PS em 60MHz, fazendo com que a taxa de serviço seja de  $1s/3,2s = 0,3125$ . As matrizes de transição do PS são mostradas em (5.3) e (5.4). Através de experimentos verificou-se que os comandos para troca de estado do OMAP 1611 através do projeto DPM são atendidos em 6ms (como mostrado na Seção 5.1) e esse tempo é desprezível em relação ao período de 1s em que a política de gerenciamento atua, o que permite modelar as transições do PS de forma determinística.

Deseja-se utilizar esse sistema durante 10000s, o que resulta em 10000 períodos e assim num fator de desconto  $\xi = 1 - 1/10000 = 0,9999$ . Deseja-se também que o número médio de requisições pendentes na FS seja de no máximo 10%, e que a perda de requisições média seja de no máximo 10%.

$$c(s, a) = \begin{matrix} & \text{trocarPara192} & \text{trocarPara60} \\ \begin{matrix} 192 \\ 60 \end{matrix} & \begin{bmatrix} 318,7 & 318,7 \\ 294,9 & 294,9 \end{bmatrix} \end{matrix} \quad (5.1)$$

$$b(s, a) = \begin{matrix} & \text{trocarPara192} & \text{trocarPara60} \\ \begin{matrix} 192 \\ 60 \end{matrix} & \begin{bmatrix} 1 & 1 \\ 0,3125 & 0,3125 \end{bmatrix} \end{matrix} \quad (5.2)$$

$$P^{PS}(\text{trocarPara192}) = \begin{matrix} & 192 & 60 \\ \begin{matrix} 192 \\ 60 \end{matrix} & \begin{bmatrix} 1 & 0 \\ 1 & 0 \end{bmatrix} \end{matrix} \quad (5.3)$$

$$P^{PS}(\text{trocarPara60}) = \begin{matrix} & 192 & 60 \\ \begin{matrix} 192 \\ 60 \end{matrix} & \begin{bmatrix} 0 & 1 \\ 0 & 1 \end{bmatrix} \end{matrix} \quad (5.4)$$

O sistema foi simulado com a política de Chung et al.(1) e com a política GDEBAT descrita na Seção 4.2. Para identificação do RS foi utilizado o método de multi-janela com comprimento 100. O resultado da identificação ao longo da simulação é mostrado na Figura 2.23. Para GDEBAT, cinco valores de restrições de penalidade de desempenho foram usadas:  $D = 0,1, 0,3, 0,5, 0,7$  e  $0,9$ . A restrição de perda de requisições foi fixada para  $L = 0,1$ . Foi usado o modelo de bateria de Rakhmatov e Vrudhula(12) com

parâmetros  $\alpha = 2838374,0484mA \cdot s$  e  $\beta = 0,0364s^{-\frac{1}{2}}$ , os mesmos identificados na Seção 5.3 para a bateria Siemens Li-Ion modelo V30145-K1310-X329 que será utilizada para esse estudo de caso. O limitante superior para o tempo de recuperação (Equação (3.22)) foi calculado para  $\varepsilon = 0,39$ , resultando em  $\bar{t}_r(0,39) = 25,168s$ . Simulou-se também o sistema com a técnica de GDE de Chung et al.(1) com  $D = 0,1$  e  $L = 0,1$  para fins de comparação. Quatro métricas foram calculadas para avaliar a qualidade das técnicas:

- I: Corrente média (mA)
- $\hat{D}$ : Número médio de requisições pendentes na FS por período (%)
- $\hat{L}$ : Perda média de requisições por período (%)
- B: Autonomia da bateria (s)

Os resultados de simulação são apresentados na Tabela 5.4. Ambas as técnicas não respeitaram a restrição de penalidade de desempenho D, o que não representa um problema dado que é de maior interesse a restrição de perda de requisições L, que é respeitada pelas duas políticas. Como esperado, a penalidade de desempenho  $\hat{D}$  foi menor para a política de Chung et al.(1) que para GDEBAT, pois a segunda troca desempenho por autonomia de bateria. A autonomia da bateria do sistema com GDEBAT foi 85s maior do que com a técnica de Chung et al.(1), ou seja, um ganho de 0,96%, que pode não ficar evidente em resultados experimentais. Esse ganho pode ser aumentado em troca da perda de desempenho modificando-se a função (4.9) sob a forma da função em (5.5), com  $k > 1$ , para que políticas otimizadas para restrições de penalidade de desempenho mais altas sejam utilizadas mais rapidamente. À semelhança de  $t_{desligado}$  na função (4.9),  $t_{60}$  representa na função (5.5) o tempo em que o OMAP permanece no ponto de operação 60.

Tabela 5.4: Comparação de algoritmos.

Algoritmo	I(mA)	$\hat{D}$	$\hat{L}$	B(s)
Chung et al.(1) ( $D = 0,1$ )	304,81	17,30%	7,50%	8849
GDEBAT ( $0,1 \leq D \leq 0,9$ )	302,10	22,34%	6,46%	8934

$$f(t_{60}) = 1 - e^{-k \cdot t_{60} / \bar{t}_r(\varepsilon)} \quad (5.5)$$

Na Tabela 5.5 são apresentados resultados de simulação de GDEBAT variando-se o parâmetro  $k$  da função (5.5). De  $k = 2$  a 6 a autonomia B da bateria aumenta em até 175s em relação à técnica de Chung et al.(1). Para  $k = 7$  o aumento de autonomia é de apenas 3s, ocorrendo aumento de perda de requisições. Para  $k = 8$  a autonomia é

menor que para  $k = 6$ . Para  $k = 9$  a autonomia aumenta apenas 6s em relação a  $k = 6$ . Para  $k = 10$  a restrição de perda de requisições já não é respeitada. Prefere-se  $k = 6s$  a  $k = 9s$  pois a constante de tempo  $\bar{t}_r(\varepsilon)/k$  da exponencial é menor: 4,19s para  $k = 6s$ , e 2,79s para  $k = 9s$ . Isso permite que o usuário do PS não sinta uma diminuição rápida de desempenho. A escolha de  $k$  permite a GDEBAT trocar desempenho por autonomia da bateria de forma controlada.

Tabela 5.5: Variando  $k$  para GDEBAT

Algoritmo	I(mA)	$\hat{D}$	$\hat{L}$	B(s)
GDEBAT ( $k = 2$ ) ( $0,1 \leq D \leq 0,9$ )	301,18	25,11%	6,96%	8961
GDEBAT ( $k = 3$ ) ( $0,1 \leq D \leq 0,9$ )	301,16	25,14%	7,01%	8961
GDEBAT ( $k = 4$ ) ( $0,1 \leq D \leq 0,9$ )	299,69	25,76%	6,88%	9006
GDEBAT ( $k = 5$ ) ( $0,1 \leq D \leq 0,9$ )	300,24	25,56%	6,57%	8989
GDEBAT ( $k = 6$ ) ( $0,1 \leq D \leq 0,9$ )	299,08	29,24%	8,11%	9024
GDEBAT ( $k = 7$ ) ( $0,1 \leq D \leq 0,9$ )	298,97	30,01%	8,61%	9027
GDEBAT ( $k = 8$ ) ( $0,1 \leq D \leq 0,9$ )	299,17	28,28%	7,99%	9021
GDEBAT ( $k = 9$ ) ( $0,1 \leq D \leq 0,9$ )	298,77	29,29%	7,48%	9033
GDEBAT ( $k = 10$ ) ( $0,1 \leq D \leq 0,9$ )	298,18	32,18%	10,68%	9051

Para avaliar experimentalmente as duas técnicas simuladas nesse estudo de caso, foram realizados 20 ensaios: 10 para GDEBAT e 10 para a técnica de Chung et al.(1). Quatro parâmetros foram obtidos:

- I: Corrente média (mA)
- $\hat{D}$ : Número médio de requisições pendentes na FS por período (%)
- $\hat{L}$ : Perda média de requisições por período (%)
- B: Autonomia da bateria (s). Após o ligamento do OMAP e a inicialização do SO Linux, a técnica de gerenciamento de energia é ativada de imediato. O tempo de medição da autonomia da bateria encerra quando o OMAP desliga devido ao fato de a bateria descarregar.

Na Figura 5.15 são mostrados os estados do sistema para o primeiro ensaio com GDEBAT. O RS gera requisições conforme (2.52) e (2.53). Os resultados da identificação do RS são ilustrados na Figura 5.16, da qual se verifica que a identificação tem desempenho equivalente ao da simulação do exemplo 10 (Figura 2.23). No início do ensaio (até 1000s) a carga de trabalho é baixa e o PS passa a maior parte do tempo no ponto de operação 60. A carga de trabalho aumenta gradativamente, obrigando o PS a permanecer cada vez

mais intervalos de tempo no ponto de operação 192 até o intervalo entre 4000s e 5000s em que a carga de trabalho se torna a mais intensa ao longo do ensaio. Em grande parte desse intervalo de tempo o PS permanece constantemente no ponto de operação 192, em que as requisições tem probabilidade 1 de serem atendidas numa fatia de tempo, e a FS fica vazia nessas condições, não havendo perda de requisições. A partir de 5000s a carga de trabalho diminui gradativamente e o sistema passa a maior parte do tempo no ponto de operação 60.

A tensão, corrente e potência que a bateria fornece ao OMAP ao longo do ensaio são mostrados na Figura 5.17. É perceptível na curva de corrente o intervalo constante de tempo em que o PS permanece no ponto de operação 192 entre 4000s e 5000s. Através da Figura 5.18 é possível observar a associação entre a corrente fornecida pela bateria e os estados assumidos pelo PS ao longo do ensaio, o que fica mais evidente na ampliação, em que aparecem os primeiros 50s de operação de GDEBAT, logo após a inicialização do Linux.

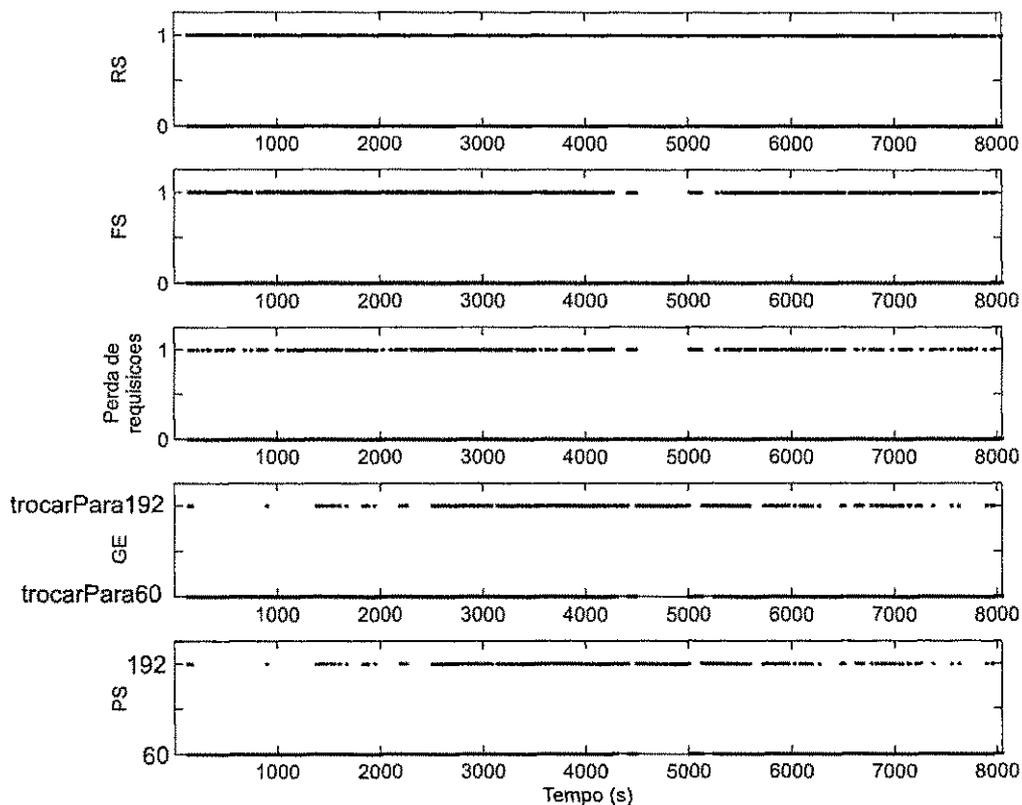


Figura 5.15: Estados do sistema ao longo do primeiro ensaio com GDEBAT.

Foram realizados dois ensaios em que o PS sempre permanece no mesmo estado: 192 ou 60. Os resultados são mostrados nas Tabelas 5.6 e 5.7. Quando o PS está sempre no estado 192, a penalidade de desempenho e a perda de requisições são nulas, pois a taxa de serviço para esse estado é 1. Quando o PS está sempre no estado 60, a penalidade

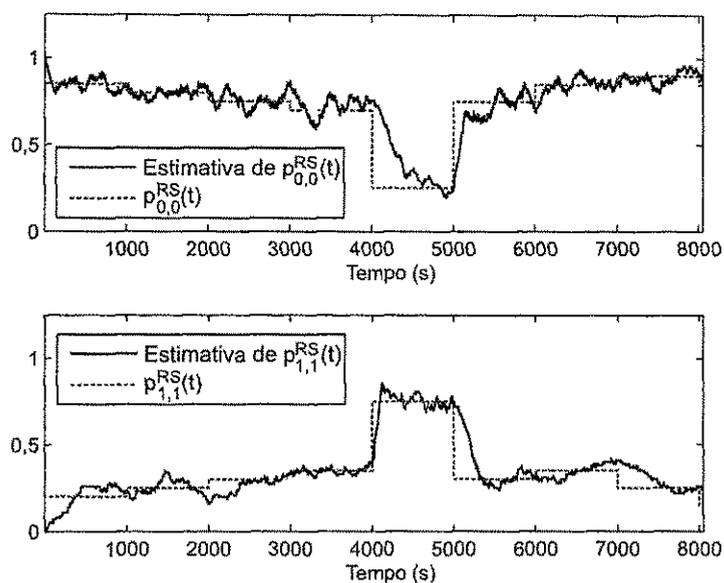


Figura 5.16: Identificação do RS ao longo do primeiro ensaio com GDEBAT.

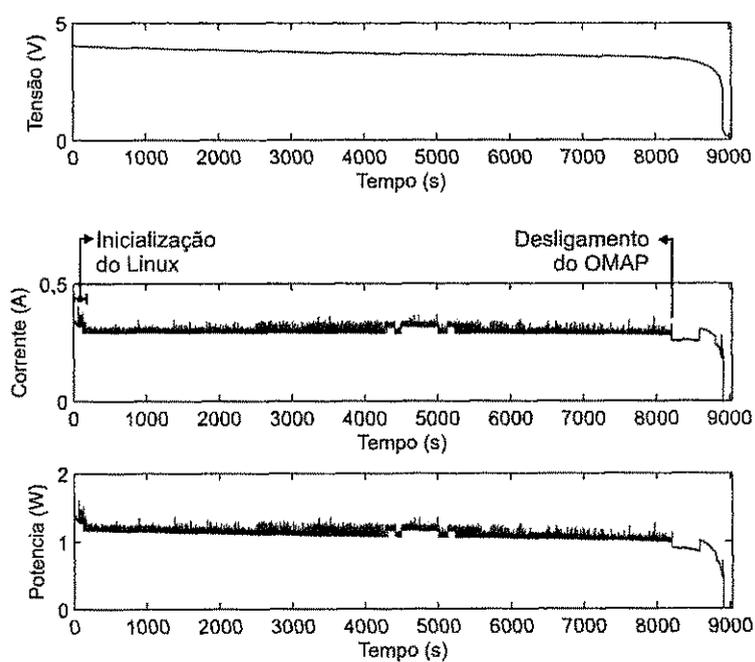


Figura 5.17: Tensão, corrente e potência da bateria ao longo do primeiro ensaio com GDEBAT.

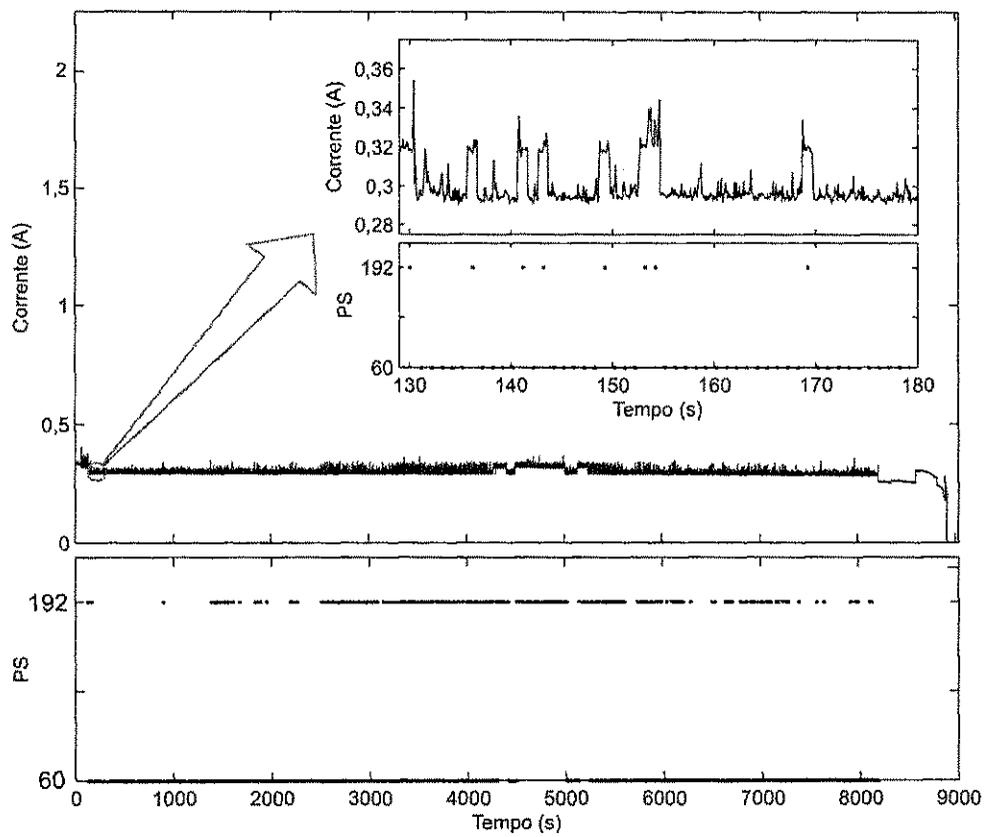


Figura 5.18: Estados do PS e corrente fornecida pela bateria ao longo do primeiro ensaio com GDEBAT, e ampliação para o intervalo de 130 a 180s.

de desempenho e a perda de requisições violam suas respectivas restrições, pois a taxa de serviço nesse estado não oferece o desempenho necessário para o sistema. Isso mostra a necessidade do uso de um GE para sejam respeitadas as restrições de penalidade de desempenho e perda de requisições. A diferença de autonomia da bateria entre os dois casos é de 1207s, que é o limite máximo de ganho que uma técnica de gerenciamento de energia pode ter nesse caso, lembrando que, se é alcançada a autonomia máxima de 9601s para o caso do estado 60, as restrições de penalidade de desempenho e perda de requisições são violadas.

Os resultados dos ensaios para GDEBAT e a técnica de Chung et al.(1) são apresentados nas Tabelas 5.8 e 5.9 respectivamente. Considerando os valores médios dessas tabelas, verifica-se que ambas as técnicas não respeitaram a restrição de penalidade de desempenho  $D$ , o que não representa um problema dado que é de maior interesse a restrição de perda de requisições  $L$ , que é respeitada pelas duas políticas. Como esperado, a penalidade de desempenho  $\hat{D}$  e a perda de requisições  $\hat{L}$  foram menores para a política de Chung et al.(1) que para GDEBAT, pois a segunda troca desempenho por autonomia de bateria. A autonomia da bateria do sistema com GDEBAT foi: 392,7s maior do que com a técnica de Chung et al.(1), ou seja, um ganho de 4,51%; e 706,9s maior do que quando o PS está sempre no estado 192 ou seja, um ganho global de 8,42%.

Tabela 5.6: Resultado experimental para ensaio do PS sempre no estado 192

I(mA)	$\hat{D}$	$\hat{L}$	B(s)
316,0	0%	0%	8394

Tabela 5.7: Resultado experimental para ensaio do PS sempre no estado 60

I(mA)	$\hat{D}$	$\hat{L}$	B(s)
289,5	40,03%	14,53%	9601

## 5.6 Conclusões

Neste capítulo foi apresentada a metodologia de implementação da política de gerenciamento de energia proposta, GDEBAT. Uma plataforma alvo e seus recursos para realização de gerenciamento de energia foram mostrados. Como GDEBAT requer a caracterização da bateria a ser usada, foi necessário medir a tensão e corrente da bateria através de um módulo de aquisição de dados e realizar ensaios a corrente constante para determinar os parâmetros  $\alpha$  e  $\beta$  da bateria. GDEBAT foi implementada na plataforma alvo e avaliada através de um estudo de caso.

Tabela 5.8: Resultados experimentais para GDEBAT

Ensaio	I(mA)	$\hat{D}$	$\hat{L}$	B(s)
1	298,6	27,65%	8,70%	8207
2	293,9	25,84%	6,77%	8727
3	278,6	27,80%	7,52%	10484
4	296,4	25,70%	5,82%	8782
5	295,9	25,80%	5,78%	8966
6	276,0	25,72%	7,00%	9103
7	279,5	26,29%	5,69%	9894
8	297,4	27,30%	7,88%	8808
9	295,4	27,20%	8,13%	8927
10	297,8	24,39%	6,64%	9111
<b>Média</b>	<b>290,95</b>	<b>26,37%</b>	<b>6,99%</b>	<b>9100,9</b>

Tabela 5.9: Resultados experimentais para a técnica de Chung et al.(1)

Ensaio	I(mA)	$\hat{D}$	$\hat{L}$	B(s)
1	307,8	10,81%	3,30%	8588
2	285,8	11,32%	3,98%	9669
3	298,6	10,94%	3,82%	8461
4	307,9	12,73%	4,76%	8973
5	301,6	10,90%	3,19%	9303
6	301,7	12,13%	4,21%	8774
7	299,7	13,34%	5,59%	8686
8	307,4	10,73%	3,72%	8092
9	311,4	11,60%	4,03%	8577
10	306,0	11,36%	4,12%	7959
<b>Média</b>	<b>302,79</b>	<b>11,59%</b>	<b>4,07%</b>	<b>8708,2</b>

## Capítulo 6

### Conclusões e trabalhos futuros

Nesse trabalho foi utilizado um modelo estocástico de sistema em processos de Markov no tempo discreto com modelo de carga de trabalho não-estacionária para resolver o problema de otimização de consumo de energia, e um modelo analítico acurado de bateria, que inclui a característica de taxa de capacidade e o fenômeno de relaxação, para considerar o comportamento da bateria no problema de gerenciamento de energia. Tomando por base esses três modelos, foi proposta uma técnica para gerenciamento dinâmico de energia orientado à autonomia da bateria, denominada GDEBAT, que se beneficia do fenômeno de relaxação da bateria para trocar de forma controlada desempenho por autonomia de bateria em sistemas móveis embarcados.

A técnica GDEBAT utiliza políticas de energia otimizadas para alto desempenho em estados de alto consumo de energia, para que o usuário do sistema experiencie alto desempenho quando efetivamente usando o sistema. E GDEBAT usa políticas de energia otimizadas para baixo desempenho em estados de baixo consumo de energia, para que, durante períodos de inatividade, o sistema permanece em estados de menor consumo de corrente, e a bateria “se recupere” através do fenômeno de relaxação. A transição entre políticas otimizadas para restrições de baixa penalidade de desempenho para políticas otimizadas para altas restrições de penalidade de desempenho é realizada suavemente, para que o usuário não experiencie um atraso grande para que o sistema retorne a um estado de alto consumo de energia e alto desempenho. Ressalta-se também que GDEBAT é capaz de lidar com cargas de trabalho não-estacionárias.

Um sistema real foi modelado e através de simulações e resultados experimentais, foi verificado que GDEBAT pode ser utilizada para trocar desempenho por autonomia da bateria de forma controlada e oferecer, na média, maior autonomia de bateria que a técnica de Chung et al.(1).

## 6.1 Trabalhos futuros

A partir do presente trabalho, podem ser derivadas várias linhas de trabalhos futuros:

- Aplicar a técnica proposta a sistema com múltiplos recursos que interagem entre si, tais como redes de sensores sem fio, que utilizam nós alimentados a bateria. O problema de otimização da política orientada a autonomia da bateria deve ser aplicada aos modelos dos recursos individuais e também à rede como um todo;
- O modelo analítico de Rakhmatov e Vrudhula(12) para estimação da autonomia da bateria é computacionalmente custoso. A partir de um perfil de descarga previamente conhecido, deve ser possível através de técnicas de cálculo computacional, reduzir consideravelmente o tempo gasto para estimar a autonomia da bateria;
- Incluir um modelo do conversor DC-DC ao modelo do sistema, para considerar a variação de eficiência e as características de consumo de energia de conversores DC-DC;
- Investigar ferramentas teóricas que possibilitem a resolução do problema de otimização da política de gerenciamento de energia usando uma função de custo de bateria, o que deve oferecer a autonomia ótima a dispositivos móveis alimentados a bateria;
- Fazer o dispositivo informar à política de gerenciamento de energia a tensão e corrente que a bateria fornece ao dispositivo em tempo de execução e investigar modelos de bateria que permitam que, com essas informações, se infira o estado de carga da bateria. Assim o problema de otimização de energia se torna um problema de malha fechada, permitindo à política de gerenciamento de energia controlar os estados de energia do sistema de forma mais eficiente, considerando o estado de carga da bateria;
- Tratar o problema da escolha da restrição de penalidade de desempenho (Seção 4.2) de forma probabilística;
- Melhorar a identificação do requisitante de serviços;
- Considerar o efeito do envelhecimento da bateria com os ciclos de uso, e a variação das características eletroquímicas em função da temperatura.

## Referências Bibliográficas

- 1 CHUNG, E.-Y. et al. Dynamic power management for nonstationary service requests. *IEEE Transactions on Computers*, IEEE Computer Society, Washington, DC, USA, v. 51, n. 11, p. 1345–1361, 2002. ISSN 0018-9340.
- 2 RONG, P.; PEDRAM, M. Battery-aware power management based on markovian decision processes. In: *Conf. Rec. of ICCAD '02*. New York, NY, USA: ACM Press, 2002. p. 707–713. ISBN 0-7803-7607-2.
- 3 LU, Y.-H.; MICHELI, G. D. Comparing system-level power management policies. *IEEE Design & Test of Computers*, IEEE Computer Society Press, Los Alamitos, CA, USA, v. 18, n. 2, p. 10–19, 2001. ISSN 0740-7475.
- 4 PALEOLOGO, G. A. et al. Policy optimization for dynamic power management. In: *Conf. Rec. of DAC '98*. New York, NY, USA: ACM Press, 1998. p. 182–187. ISBN 0-89791-964-5.
- 5 UHRIG, S.; UNGERER, T. Hardware-based power management for real-time applications. *Second International Symposium on Parallel and Distributed Computing*, IEEE Computer Society, p. 258–265, 2003.
- 6 SOFTWARE, I. A. R. L. e M. *Dynamic Power Management for TI OMAP*. Maio 2007. Acessado em 25 de Fevereiro de 2008. Disponível em: <<http://dynamicpower.sourceforge.net/dpm-omap.html>>.
- 7 LUIZ, S. O. D. et al. A methodology to deploy applications on the dual-core omap platform. In: *Conf. Rec. of CBA 2006*. Salvador, BA, Brasil: Anais do CBA 2006, 2006. p. 3360–3365.
- 8 GORGÔNIO, U. S. et al. Application profiling in a dual-core platform. In: *Conf. Rec. of ICCE 2008*. Piscataway: IEEE Consumer Electronics Society, 2008.
- 9 LUIZ, S. O. D. et al. Formal specification of dsp gateway for data transmission between processor cores of omap platform. In: *Conf. Rec. of ACM SAC'08*. New York: ACM, 2008.

- 10 SIMUNIC, T. et al. Dynamic voltage scaling and power management for portable systems. In: *Conf. Rec. of DAC '01*. New York, NY, USA: ACM Press, 2001. p. 524–529. ISBN 1-58113-297-2.
- 11 RAKHMATOV, D. N.; VRUDHULA, S. B. K. An analytical high-level battery model for use in energy management of portable electronic systems. In: *Conf. Rec. of ICCAD '01*. Piscataway, NJ, USA: IEEE Press, 2001. p. 488–493. ISBN 0-7803-7249-2.
- 12 RAKHMATOV, D.; VRUDHULA, S. Energy management for battery-powered embedded systems. *ACM Transactions on Embedded Computing Systems*, ACM Press, New York, NY, USA, v. 2, n. 3, p. 277–324, 2003. ISSN 1539-9087.
- 13 BENINI, L. et al. Battery-driven dynamic power management. *IEEE Design & Test*, IEEE Computer Society Press, Los Alamitos, CA, USA, v. 18, n. 2, p. 53–60, 2001. ISSN 0740-7475.
- 14 ZHONG, L.; JHA, N. K. Dynamic power optimization of interactive systems. In: *Conf. Rec. of VLSI '04*. Washington, DC, USA: IEEE Computer Society, 2004. p. 1041. ISBN 0-7695-2072-3.
- 15 ANAND, M.; NIGHTINGALE, E. B.; FLINN, J. Ghosts in the machine: interfaces for better power management. In: *Conf. Rec. of MobiSys '04*. New York, NY, USA: ACM Press, 2004. p. 23–35. ISBN 1-58113-793-1.
- 16 VINIOTIS, Y. *Probability and random processes for electrical engineers*. Boston: WCB/McGraw-Hill, 1998. xviii + 676 p. (McGraw-Hill series in electrical and computer engineering. Communications and signal processing). ISBN 0-07-067491-4.
- 17 MEYER, P. L. *Probabilidade: aplicações à estatística*. 2. ed. Rio de Janeiro: LTC - Livros Técnicos e Científicos Editora S.A., 1983. ISBN 85-216-0294-4.
- 18 CZYZYK, J. et al. *PCx User Guide (Version 1.1)*. Novembro 1997. Acessado em 29 de Fevereiro de 2008. Disponível em: <[www.unix.mcs.anl.gov/otc/Tools/PCx/doc/PCx-user.pdf](http://www.unix.mcs.anl.gov/otc/Tools/PCx/doc/PCx-user.pdf)>.
- 19 MATHWORKS, I. T. *linprog*. Fevereiro 2008. Acessado em 29 de Fevereiro de 2008. Disponível em: <<http://www.mathworks.com/access/helpdesk/help/toolbox/optim/ug/linprog.html>>.
- 20 LINDEN, D.; REDDY, T. B. *Handbook of batteries*. 3. ed. [S.l.]: McGraw-Hill, 2001. ISBN 0-07-135978-8.

- 21 CHIASSERINI, C. F.; RAO, R. R. Pulsed battery discharge in communication devices. In: *Conf. Rec. of MobiCom '99*. New York, NY, USA: ACM, 1999. p. 88–95. ISBN 1-58113-142-9.
- 22 NEWMAN, J. et al. Modeling of lithium-ion batteries. Elsevier, v. 119-121, p. 838–843, 2003.
- 23 CHEN, M.; RINCÓN-MORA, G. A. Accurate electrical battery model capable of predicting runtime and i-v performance. *Energy Conversion, IEEE Transaction on*, v. 21, n. 2, p. 504–511, June 2006. ISSN 0885-8969.
- 24 PANIGRAHI, D. et al. Battery life estimation of mobile embedded systems. *Conf. Rec. of VLSI Design, 2001*, p. 57–63, 2001.
- 25 RAO, R.; VRUDHULA, S. Battery optimization vs energy optimization: which to choose and when? In: *Conf. Rec. of ICCAD '05*. Washington, DC, USA: IEEE Computer Society, 2005. p. 439–445. ISBN 0-7803-9254-X.
- 26 INSTRUMENTS, T. *High-Performance: OMAP1611*. Janeiro 2008. Acessado em 25 de Fevereiro de 2008. Disponível em: <<http://focus.ti.com/general/docs/wtbu/wtbuproductcontent.tsp?templateId=6123&navigationId=11993&contentId=4668>>.
- 27 VISTA, M.; IBM. *Dynamic power management for embedded systems*. Acessado em 25 de Fevereiro de 2008. Disponível em: <[citeseer.ist.psu.edu/567075.html](http://citeseer.ist.psu.edu/567075.html)>.
- 28 BROCK, B.; RAJAMANI, K. Dynamic power management for embedded systems. In: *Conf. Rec. of SOC Conference, 2003*. Portland, Oregon: Proceedings of the IEEE International SOC Conference, 2003. p. 416–419. ISBN 0-89791-964-5.
- 29 SOFTWARE, I. A. R. L. e M. *dpm-utils-omap package*. Maio 2007. Acessado em 25 de Fevereiro de 2008. Disponível em: <<http://ufpr.dl.sourceforge.net/sourceforge/dynamicpower/dpm-utils-omap.tar.gz>>.
- 30 PERCIVAL, M. *OSK5912 General Purpose Timers*. Agosto 2005. Acessado em 25 de Fevereiro de 2008. Disponível em: <[www.capgo.com/Resources/SoftwareDev/osk-gptimers.pdf](http://www.capgo.com/Resources/SoftwareDev/osk-gptimers.pdf)>.