



UNIVERSIDADE FEDERAL DO AMAPÁ – UNIFAP
BACHARELADO EM ENGENHARIA ELÉTRICA

CAIO SOARES MODESTO

**AVALIAÇÃO DE UM CONTROLADOR PSO EM SISTEMA FOTOVOLTAICO DE
BOMBEAMENTO**

MACAPÁ - AP

2019

CAIO SOARES MODESTO

**AVALIAÇÃO DE UM CONTROLADOR PSO EM SISTEMA FOTOVOLTAICO DE
BOMBEAMENTO**

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia Elétrica da Universidade Federal do Amapá – UNIFAP, como requisito para a obtenção do grau de bacharel em Engenharia Elétrica.

Orientador: Dr. Geraldo Neves de A. Maranhão.

MACAPÁ - AP

2019

Dados Internacionais de Catalogação na Publicação (CIP)
Biblioteca Central da Universidade Federal do Amapá
Elaborada por Cristina Fernandes – CRB-2/1569

Modesto, Caio Soares.

Avaliação de um controlador PSO em sistema fotovoltaico de bombeamento. / Caio Soares Modesto ; orientador, Geraldo Neves de Albuquerque Maranhão. – Macapá, 2019.

73 f.

Trabalho de Conclusão de Curso (Graduação) – Fundação Universidade Federal do Amapá, Coordenação do Curso de Bacharelado em Engenharia Elétrica.

1. *Particle Swarm Optimization*. 2. Sistema fotovoltaico de bombeamento. 3. Controle adaptativo. I. Maranhão, Geraldo Neves de Albuquerque, orientador. II. Fundação Universidade Federal do Amapá. III. Título.

333.7 M691a
CDD: 22. ed.

CAIO SOARES MODESTO

AVALIAÇÃO DE UM CONTROLADOR PSO EM SISTEMA FOTOVOLTAICO DE BOMBAMENTO

Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Engenharia Elétrica da Universidade Federal do Amapá como requisito à obtenção do grau de título de Engenheiro Eletricista.

Orientador: Dr. Geraldo Neves de A. Maranhão.

BANCA AVALIADORA

Orientador: Prof. Dr. Geraldo Neves de Albuquerque Maranhão

Prof. Dr. Alaan Ubaiara Brito

Avaliador/UNIFAP

Prof. Me. Raphael Diego Comesanha e Silva

Avaliador/UNIFAP

Nota: _____.

Data da Aprovação: Macapá, _____ de _____ de 2019.

AGRADECIMENTOS

Agradeço primeiramente a Deus por tudo, à minha família e amigos por serem parte da minha vida.

Agradeço aos professores, que além de toda a carga simbólica que representam como fontes de conhecimento, demonstraram a dedicação necessária para que essa etapa da minha vida se concretizasse.

Agradeço aos demais que participaram ou contribuíram de alguma forma para o andamento deste trabalho.

RESUMO

No âmbito do controle de processos, com ênfase na busca por diversificação de métodos de controle, este trabalho apresenta o *Particle Swarm Optimization* para a realização de um controle adaptativo direto de um Sistema fotovoltaico de bombeamento, na busca pelo melhor aproveitamento energético de fontes intermitentes de energia. Sendo que tal técnica de otimização se destaca das demais por sua simplicidade de programação e velocidade de processamento, devido ao seu curto algoritmo e fácil parametrização. Sendo assim, objetiva-se aplicar um sinal de controle em um Conversor de Frequência comercial através de Dispositivo de Programação de Controladores e Monitoramento, de modo a tornar constante a tensão de operação de um sistema fotovoltaico de bombeamento através do controle de frequência de uma motobomba. Dessa forma, para a produção do sinal de controle é utilizado o algoritmo em questão, de modo a aproveitar sua característica de otimização para minimizar o erro entre a tensão c.c. de alimentação pelo Gerador Fotovoltaico e o *setpoint*. Para tanto, será utilizado como metodologia o uso de bancada de ensaios do referido sistema, disponibilizada pela Universidade Federal do Amapá, a qual apresenta os equipamentos necessários para aplicação deste trabalho. Foram obtidos resultados favoráveis para a proposta de uso do *Particle Swarm Optimization* como método de controle adaptativo, contudo, o controlador se mostrou pouco robusto para as variações bruscas de potência fornecida pelo Gerador Fotovoltaico, característica do próprio sistema.

Palavras-chave: *Particle Swarm Optimization*. Sistema fotovoltaico de bombeamento. Controle adaptativo.

ABSTRACT

In the scope of process control, with emphasis on the search for diversification of control methods, this work presents the Particle Swarm Optimization to perform a direct adaptive control of a photovoltaic pumping system, in the search for better energy utilization of intermittent sources of energy. Since such optimization technique stands out from the others for its simplicity of programming and processing speed, due to its short algorithm and easy parameterization. Therefore, it is intended to apply a control signal in a commercial frequency converter through a Controller and Monitor Programming Device, in order to make constant the operating voltage of a pumping system by means of the frequency control of a motor pump. Thus, for the production of the control signal the algorithm in question is used, so as to take advantage of its optimization characteristic to minimize the error between the voltage c.c. by the Photovoltaic Generator and the setpoint. To do so, it will be used as methodology the use of test bench of said system, made available by the Federal University of Amapá, which presents the necessary equipment to apply this work. We obtained favorable results for the proposed use of Particle Swarm Optimization as a method of adaptive control, however, the controller was not very robust for the sudden variations of power provided by the Photovoltaic Generator, characteristic of the system itself.

Keywords: Particle Swarm Optimization. Photovoltaic pumping system. Adaptive control

LISTA DE FIGURAS

FIGURA 01 - Bando de pássaros em voo.....	14
FIGURA 02 - Trajetória de movimento de uma partícula.....	15
FIGURA 03 - Curva I x V de um módulo fotovoltaico.....	20
FIGURA 04 - Curva I x V para a associação de módulos fotovoltaicos.....	21
FIGURA 05 - Influência da variação da irradiância na curva I x V de uma célula fotovoltaica de silício cristalino à temperatura de 25°C.....	22
FIGURA 06 - Efeito causado pela temperatura das células na curva I x V (para 1000w/m) em um módulo fotovoltaico.....	22
FIGURA 07 - Efeito do sombreamento em um arranjo fotovoltaico, curva P-V.....	23
FIGURA 08 - Diagrama elétrico básico de um conversor de frequência convencional.....	24
FIGURA 09 - Fluxograma da lógica de controle do microcontrolador Escravo.....	29
FIGURA 10 - Fluxograma da lógica de controle do microcontrolador Mestre.....	30
FIGURA 11 - Diagrama esquemático da bancada de ensaios.....	33
FIGURA 12 - Foto dos módulos fotovoltaicos em cima do Prédio do Bloco de Engenharia Elétrica.....	34
FIGURA 13 - Foto da bancada de ensaios de SFB no laboratório da Universidade Federal do Amapá com a identificação dos seus componentes.....	35
FIGURA 14 - Diagrama de blocos simplificado do sistema de controle.....	38
FIGURA 15 - Irradiância x Potência entregue à motobomba com controlador PSO.....	43
FIGURA 16 - Corrente entregue pelo CF à motobomba.....	44
FIGURA 17 - Tensão no barramento c.c. do CF com a atuação do controlador PSO.....	44
FIGURA 18 - Gráficos de potência x Irradiância e tensão, entre 12:30 e 14:26 horas.....	45
FIGURA 19 - Gráficos de potência x Irradiância e tensão, entre 14:26 e 15:50 horas.....	46
FIGURA 20 - Gráficos de potência x Irradiância e tensão, entre 15:50 e 17:46 horas.....	47

LISTA DE TABELAS

Tabela 01 - Características físicas e elétricas da motobomba em teste.....	34
Tabela 02 - Características do Módulo Fotovoltaico utilizado.....	35
Tabela 03 - Ficha técnica do DPCM.....	37
Tabela 04 - Resumo das características técnicas dos componentes do sistema de aquisição de dados.....	37
Tabela 05 - Quantidade de interrupções por subtensão e horários em horas.....	47

LISTA DE ABREVIATURAS E SIGLAS

ANFIS - *adaptive neuro-fuzzy inference system*

CF- Conversor de Frequência

DPCM - Dispositivo de Programação de Controladores e Monitoramento

FV - Fotovoltaico

GFV - Gerador Fotovoltaico

GWO - *Grey Wolf Optimizer*

IDE - *Integrated Development Environment*

PID - Proporcional Integral Derivativo

PSO - *Particle Swarm Optimization*

PWM - *Pulse Width Modulation*

SFB - Sistema Fotovoltaico de Bombeamento

UNIFAP - Universidade Federal do Amapá

SUMÁRIO

1 INTRODUÇÃO.....	11
2 FUNDAMENTAÇÃO TEÓRICA.....	13
2.1 Particle swarm optimization	13
<i>2.1.1 PSO clássico.....</i>	<i>14</i>
<i>2.1.2 Constante de inércia e fator de contração.....</i>	<i>16</i>
<i>2.1.3 As condições de restrição.....</i>	<i>17</i>
2.2 Sistema fotovoltaico de bombeamento em conexão direta.....	18
<i>2.2.1 Gerador fotovoltaico.....</i>	<i>20</i>
<i>2.2.2 Conversor de frequência.....</i>	<i>24</i>
<i>2.2.3 Motobomba.....</i>	<i>25</i>
2.3 Controle dinâmico de sistema.....	26
<i>2.3.1 Controle adaptativo de sistemas dinâmicos.....</i>	<i>28</i>
<i>2.3.2 Dispositivo de programação de controladores e monitoramento.....</i>	<i>29</i>
3 MATERIAIS E MÉTODOS.....	32
3.1 Bancada de ensaios de SFB.....	32
<i>3.1.1 DPCM aplicado à bancada de ensaios.....</i>	<i>36</i>
3.2 Implementação do controlador PSO no SFB.....	37
<i>3.2.1 Código do controlador PSO.....</i>	<i>39</i>
4 RESULTADOS.....	43
5 CONCLUSÃO.....	49
REFERÊNCIAS.....	50
APÊNDICE A – Algoritmo para programação do Microcontrolador Mestre.....	53
APÊNDICE B – Algoritmo para programação do Microcontrolador Escravo.....	59
APÊNDICE C – Simulação de controlador PSO proporcional em motor c.c.....	66

1 INTRODUÇÃO

O controle de processos e sistemas é assunto de grande relevância na área da Engenharia, estando o campo de projeto de controladores analógicos proporcional integral derivativo (PID) já consolidado quando se trata de sistemas lineares, conforme se observa pela obra de Ogata (2002). Contudo, de acordo com o estudo de Serapião (2009), existem casos de sistemas que exigem maior adequação para seu controle: quando estes apresentam interrupções, não linearidade ou quaisquer dificuldades que tornem o controle convencional ineficiente. Nesses casos, são desenvolvidas técnicas com diferentes abordagens de controle, de modo a melhor atender os referidos sistemas.

Técnicas de otimização vêm sendo aprimoradas de forma eficaz para seu uso em problemas computacionais. De fato, conforme apresentado por Sun, Lai e Wu (2012), verificam-se diversas formas de modificações nas técnicas de otimização, resultando nas mais variadas características, como para funções com múltiplos máximos e mínimos, vastos intervalos de busca das variáveis independentes e outras complicações que requerem maior capacidade de otimização.

Dentre os métodos baseados em inteligência de enxames se destaca o *Particle Swarm Optimization* (PSO) pela sua simplicidade de programação e boa capacidade de convergência, já que a base do algoritmo são as equações de posição e velocidade das partículas, com poucos parâmetros a serem avaliados. Dessa forma, o PSO mostra pontos vantajosos no âmbito do controle (SERAPIÃO, 2009).

Verifica-se por produções acadêmicas da área de engenharia que há difusão do uso do PSO para auxílio de técnicas de controle, onde a planta a ser controlada é submetida a testes sob o uso do PSO para se buscar o comportamento do sistema, os melhores valores de ganhos e melhor parametrização do controlador; seja para o controle clássico, como no estudo de Lopes (2017), sintonizando os ganhos do controlador PID; seja ajustando ganhos de controlador baseado em inteligência artificial, como o trabalho de Al-Dunainawi, Abbod e Jizany (2017).

Nesse sentido, verifica-se pelo trabalho de Sun, Lai e Wu (2012) que o algoritmo do PSO recebeu inúmeros aprimoramentos conforme desenvolveu-se seu estudo, mas conforme abordado, seu uso no campo do controle ainda se restringe ao auxílio de outras técnicas e não havendo o seu uso direto como controle adaptativo.

Ademais, o Sistema Fotovoltaico de Bombeamento (SFB) se apresenta com grande relevância em caráter social ao atender localidades não eletrificadas, com o uso de sistemas de conexão direta entre um gerador fotovoltaico (GFV) e a carga alimentada, a motobomba.

Contudo, existe a necessidade de se adaptar a capacidade de geração intermitente do gerador à carga, para se obter melhor aproveitamento energético (MARANHÃO, 2015).

A admissão de técnicas de controle para sistemas dos mais variados é importante para a diversificação do estudo em controle, tanto para a realização de estudo comparativo em relação a outros métodos de controle, visando averiguar as técnicas mais apropriadas para cada caso, quanto para o avanço do uso e inovação (SERAPIÃO, 2009).

Em decorrência do exposto, este trabalho propõe o uso de SFB com controle adaptativo provido pelo PSO, de modo a tornar tensão de operação do GFV fixa, *setpoint* fixo de tensão, na operação da motobomba, considerando a variação de potência no fornecimento de energia pelo GFV. Dessa forma, o presente estudo visa apresentar um controlador PSO adaptativo, de modo que o controle é aplicado no Conversor de Frequência (CF) que atua na planta que compõe o SFB através do Dispositivo de Programação de Controladores e Monitoramento (DPCM), utilizando para realização dos testes práticos a bancada de ensaios de sistemas fotovoltaicos de bombeamento de água do curso de Engenharia Elétrica da Universidade Federal do Amapá (UNIFAP).

Este estudo tem como objetivos específicos: introduzir o PSO como método de controle adaptativo, abordagem diferenciada de seu atual uso em controle; analisar seu comportamento para a aplicação proposta; e parametrizar o algoritmo do PSO de modo a obter o melhor comportamento.

Este trabalho está estruturado em três principais capítulos, dos quais o capítulo 2 busca trazer o estado da arte sobre os principais assuntos abordados com a fundamentação teórica deste trabalho, com a apresentação e descrição das características e principais conceitos sobre o PSO, o SFB, e sobre o controle dinâmico de sistemas em conjunto com o DPCM, de modo a delimitar os conhecimentos mais relevantes para este estudo. Já o capítulo 3, Materiais e métodos, fará a descrição da bancada de ensaios do SFB provido do DPCM e da aplicação do algoritmo, descrevendo a metodologia utilizada neste trabalho. Nesse sentido, o capítulo 4 apresentará os resultados já obtidos a título de estudo introdutório, com a simulação do PSO para controle de motor c.c., e os resultados da proposta deste estudo. Por fim, as considerações finais do trabalho apresentarão um panorama geral sobre o trabalho e demais ponderações.

2 FUNDAMENTAÇÃO TEÓRICA

Com o intuito de apresentar os embasamentos necessários para as principais áreas deste estudo, o presente capítulo apresenta os fundamentos sobre a técnica de otimização por enxame de partículas, o sistema fotovoltaico de bombeamento e o controle dinâmico de sistemas. Essa abordagem permeia os principais conhecimentos da proposta deste trabalho, que é usar um controlador PSO no SFB.

O item 2.1 apresenta o PSO, em sua formulação inicial, bem como os elementos de seu algoritmo e demais conceitos necessários à sua compreensão, com propostas de modificações que resultem em melhor desempenho para aplicação em controle.

O item 2.2 delimita o SFB abordando seus componentes que se mostram importantes para a devida realização deste trabalho, que são: seu elemento de geração de energia, o GFV, é parte fundamental da proposta pois é o que justifica o uso de um sistema de controle que dê melhor respostas às bruscas variações no despacho de potência; o CF, dispositivo através do qual é aplicado o controle; e a motobomba, pois seu entendimento e comportamento são necessários para a correto funcionamento da proposta.

Por fim, o item 2.3, norteia o controle dinâmico, a fim de justificar o uso do PSO, concebido para otimização, como método de controle. Sendo assim, é abordado de forma introdutória a metodologia de controle dinâmico por controle clássico e adaptativo, e é apresentado o DPCM como dispositivo auxiliar na implementação dessas técnicas.

2.1 *Particle swarm optimization*

No âmbito do desenvolvimento de técnicas de otimização, destaca-se o crescimento da inteligência de enxames, com o desenvolvimento de diversos algoritmos e suas aplicações resultantes, com ampla gama de aplicações na engenharia. Tais métodos são definidos por técnicas matemáticas e objetivam buscar o valor máximo, mínimo ou ótimo correspondente em funções ou problemas (EBERHART; SHI, 2001).

Sendo uma técnica de otimização meta-heurística (paradigma de busca baseados em comportamentos observáveis), o PSO supre uma grande deficiência dos métodos de controle convencionais, que é a otimização sem o conhecimento do total comportamento do sistema, visto que a busca do melhor resultado pelo PSO independe desse conhecimento, sendo assim, podendo ser adotado para o controle de sistemas dinâmicos, não lineares ou com interações complexas (SERAPIÃO, 2009).

O método de otimização por enxame de partículas foi introduzido por Kennedy e Eberhart (1995) com as principais características de baixa necessidade de esforço computacional, menor necessidade de uso de memória e alta capacidade de convergência.

Essas características provêm de seu curto algoritmo, que utiliza operações matemáticas básicas, e que propõe a ideia de modelar o movimento de um grupo que busca um objetivo, com base em equações de posição e velocidade dos indivíduos desse grupo, isso gera um algoritmo com poucos parâmetros a serem avaliados, fazendo com que sua implementação e parametrização sejam simplificadas (SUN; LAI; WU, 2012).

2.1.1 PSO clássico

O desenvolvimento de técnicas meta-heurísticas se popularizou não só na área de Ciências da Computação como também em outras áreas do conhecimento, como engenharias e economia. Este fato se deve, dentre outros, à alta capacidade de busca pelo melhor resultado global de um problema; destacando-se tanto métodos antigos – como o próprio PSO, quanto métodos mais atuais, como o *Grey Wolf Optimizer* (GWO), meta-heurística bioinspirada baseada no modo de caça de um grupo de lobos (MIRJALILI; MIRJALILI; LEWIS, 2014).

Concebido inicialmente para otimização de funções contínuas não lineares, o PSO apresenta a modelagem matemática do comportamento social simplificado de grupos como um bando de pássaros ou um cardume de peixes na movimentação para evitar predadores ou buscar alimento (KENNEDY; EBERHART, 1995). Conforme apresentado na figura 01.

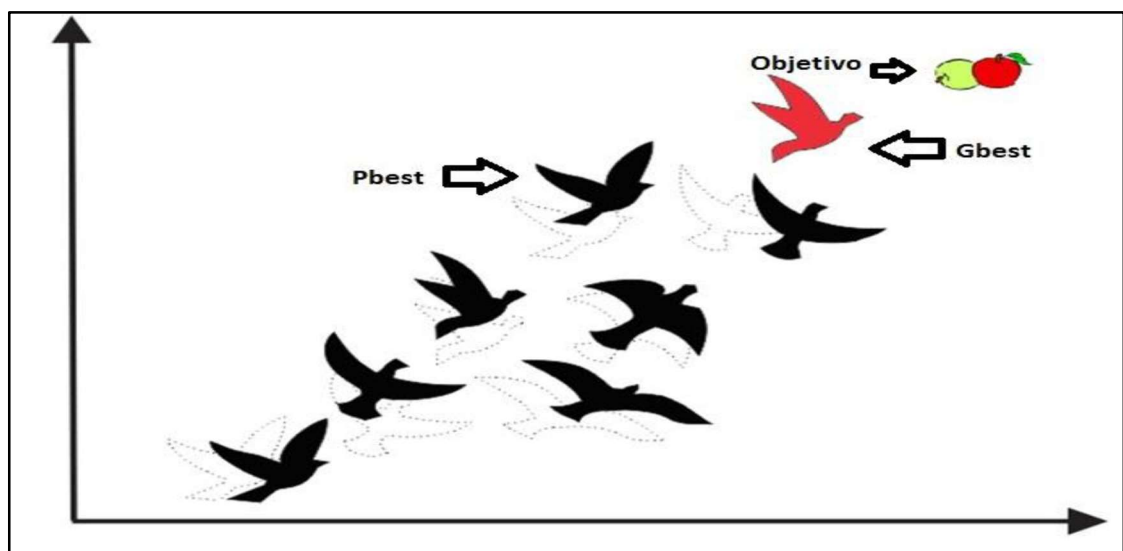


Figura 01: Bando de pássaros em voo

Fonte: GARCIA; ONAKA (2016)

Dessa forma, consideram-se os indivíduos de um bando como partículas presentes no espaço, estas se movimentam em direção a um objetivo, seu movimento é influenciado pelos seus conhecimentos prévios – sua própria cognição – e pelo conhecimento compartilhado pelo grupo – a informação compartilhada (KENNEDY; EBERHART, 1995).

Com base no que é apresentado por Sun, Lai e Wu (2012), compreende-se que sendo x_n a posição vetorial de uma partícula em um espaço n-dimensional, em que i é o valor da iteração na qual aquela se encontra e v_n sua velocidade vetorial, a partícula assume novos valores se somando com a velocidade.

A velocidade vetorial da partícula, por sua vez, também modifica seus valores, somando-se a dois componentes vetoriais, chamadas de cognitiva e social. Tais componentes estão simbolicamente representadas na figura 01 como $Pbest$ e $Gbest$, respectivamente, sendo apresentado a seguir a descrição tanto destas componentes como dos demais elementos do código.

Sendo assim, a componente cognitiva é aquela composta pela diferença vetorial do valor assumido pela partícula com o valor que lhe resultou em melhor avaliação (*fitness*), $pbest_n^i$; essa diferença é acrescida do coeficiente C , chamado de coeficiente *cognitivo*, e de um randômico entre zero e um; essa componente é chamada cognitiva por direcionar a partícula para mais próximo de sua melhor avaliação, logo, baseia-se em sua cognição (SUN; LAI; WU, 2012).

Além disso, a componente social é similar à cognitiva só que levando a partícula para mais próximo do valor mais bem avaliado dentre todas, $Gbest^i$, gerando a interação entre todo o grupo, por isso chamada de social; esse outro valor é acrescido do coeficiente S , chamado de coeficiente *social*, e de um randômico entre zero e um (SUN; LAI; WU, 2012).

A forma iterativa da posição e velocidade das partículas estão apresentadas nas equações abaixo:

$$x_n^{i+1} = x_n^i + v_n^{i+1} \quad (2.1)$$

$$v_n^{i+1} = v_n^i + (C * rand * (pbest_n^i - x_n^i)) + (S * rand * (Gbest^i - x_n^i)) \quad (2.2)$$

Conforme demonstra a equação acima, o movimento vetorial de uma partícula é melhor entendido com a Figura 02 que mostra a trajetória e influências vetoriais das componentes do movimento.

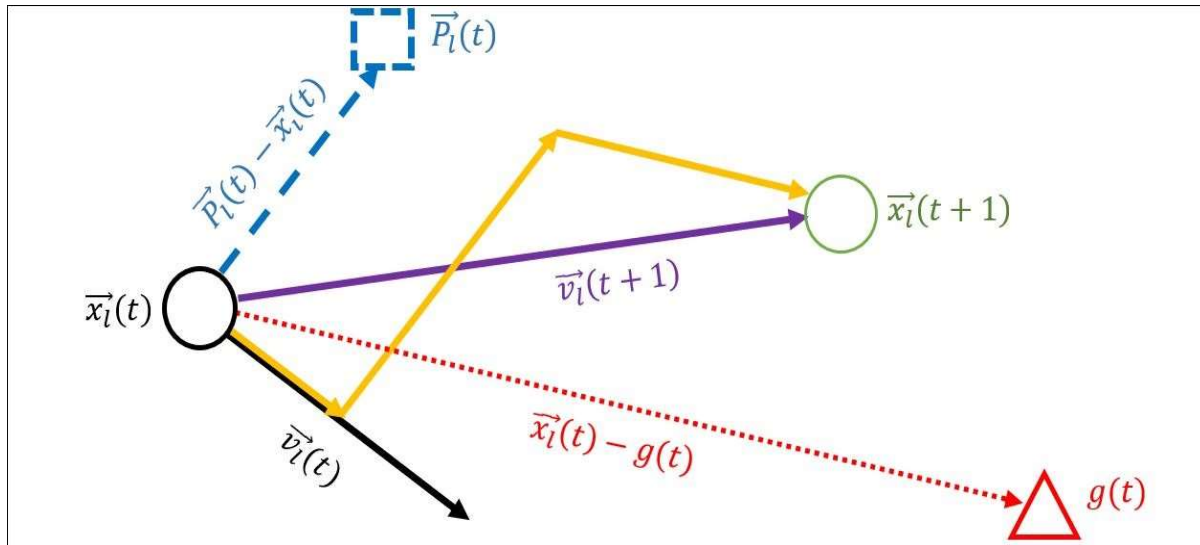


Figura 02: Trajetória do movimento de uma partícula

Fonte: Elaborado pelo autor

Sucintamente, pode ser descrito o algoritmo base do PSO como a modificação do valor das partículas x_n através do movimento destas no espaço n-dimensional através de suas velocidades, calculadas com base em suas avaliações na função objeto de otimização, de modo que as partículas buscam valores que correspondem à otimização desejada.

Tal algoritmo inicialmente desenvolvido necessitava de aprimoramentos para seu desenvolvimento e consolidação como método eficaz de otimização. Sendo assim, serão introduzidos técnicas e conceitos que supriram essa necessidade, e que são cruciais para tornar esta técnica a melhor possível para uso no controle de processos.

2.1.2 Constante de inércia e fator de restrição

De acordo com o estudo de Shi e Eberhart (1998), notou-se que o algoritmo original do PSO tinha dificuldades de convergência relacionadas ao cálculo de sua velocidade, pois quando a posição das partículas atingia valores próximos do desejado, o módulo da velocidade, que é somado à posição da partícula, estava em valores elevados, resultando em grandes variações espaciais da posição das partículas que neste ponto deveriam diminuir suas velocidades e se aproximar do ponto desejado de maneira mais rápida.

Para melhorar o cálculo da velocidade em relação à sua inércia, que é a consideração do valor anterior da velocidade em sua atualização, é utilizada a constante de inércia (w), a fim de limitar o valor acumulado de velocidade anterior da partícula e com isso podendo melhorar a

influência dos fatores cognitivo e social da partícula, resultando em convergências mais rápidas (SHI; EBERHART, 1998).

É abordado uma constante de inércia fixa, diferentemente do que foi inicialmente proposto por Shi e Eberhart (1998), que abordaram esse elemento com variação por iteração, já que tal otimização foi campo de estudo de ciências computacionais, que abordam problemas simulados. Portanto, a equação de velocidade acrescida da constante de inércia fica conforme a equação abaixo:

$$v_n^{i+1} = w * v_n^i + (C * rand * (pbest_n^i - x_n^i)) + (S * rand * (Gbest^i - x_n^i)) \quad (2.3)$$

Essa abordagem de parâmetro fixo é crucial para o problema Real, já que a atuação do método não é por iteração, mas sim contínua e indeterminada ao longo do tempo, perdendo parte da eficiência da constante de inércia.

Contudo, outra forma de melhorar a convergência, muito eficiente no algoritmo do PSO, é a utilização do fator de constrição (EBERHART; SHI, 2000).

$$Fc = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}, \varphi = C + S, \varphi > 4 \quad (2.4)$$

Dessa forma, o cálculo da velocidade de uma partícula, com o uso tanto do fator de constrição quanto da constante de inércia é apresentado abaixo:

$$v_n^{i+1} = Fc * [w * v_n^i + (C * rand * (pbest_n^i - x_n^i)) + (S * rand * (Gbest^i - x_n^i))] \quad (2.5)$$

Por Eberhart e Shi (2000) foi verificado que o fator de constrição em relação ao coeficiente de inércia gera melhores convergências para quando há limitação da velocidade. Contudo, no algoritmo modificado do PSO utilizado no trabalho de Maranhão e Modesto (2017) são utilizados ambos os fatores, pois verificou-se que mesmo com o fator de constrição a variação do fator de inércia influencia a característica de resposta de modo que é possível aproveitá-la.

2.1.3 As condições de restrição

O estudo de Eberhart e Shi (2000) observa que para as funções nas quais as partículas não encontram resultados ótimos de posição há um fenômeno de dispersão entre elas que, para certas funções, podem resultar em divergência do valor ideal ou até instabilidade da resposta.

Dessa forma, além do incremento do fator de restrição e constante de inércia, anteriormente abordados, é proveitoso que se façam outras restrições à movimentação das partículas.

Para que se resolva este tipo de situação é necessário restringir a posição e velocidade das partículas para que não haja divergência ou migração à instabilidade. Conforme a análise realizada por Shi et al (2001) de várias propostas de aplicação do PSO, verificou-se que é melhorada a acomodação quando se adota a limitação dos valores de exploração ou quando são definidas as equações que determinam a movimentação das partículas.

Sendo assim, para obtenção dos resultados descritos no Apêndice C deste estudo, foi utilizado para impedimento de dispersão das partículas o valor do seu desvio padrão, sendo determinado um valor que serve para estipular o quão longe cada uma poderia se afastar do grupo.

Para tanto, caso haja afastamento maior que o valor da média somado ao desvio padrão elevado a um valor decimal a partícula é deslocada para a média e sua velocidade é recalculada, conforme apresentado no Apêndice C. Desta forma, a parametrização pode ser modificada dependendo do comportamento que se quer das partículas, mais espalhadas ou contidas (MARANHÃO; MODESTO, 2017).

Foi tratado tanto no subitem 2.1.2 quanto neste que o PSO recebeu diversas modificações e melhorias, sendo assim, é necessário atentar para quais desses aprimoramentos utilizar, sendo abordadas apenas algumas alterações que se mostram bastante vantajosas no uso desta técnica.

Logo, com a devida abordagem dos principais pontos do PSO, faz-se necessário discorrer sobre os demais conhecimentos necessários à delimitação deste trabalho: o sistema fotovoltaico de bombeamento, a ser trabalhado no item 2.2; e o controle dinâmico, no item 2.3.

2.2 Sistema Fotovoltaico de Bombeamento em Conexão Direta

Uma das formas de utilização da energia solar é a sua transformação em energia elétrica pelo efeito fotovoltaico, por meio de geradores fotovoltaicos (GFV), nesse sentido, objetiva-se apresentar os principais conceitos associados a um sistema de conexão direta e destinado a bombeamento, o SFB.

Tais sistemas podem estar presentes em áreas urbanas, mas conforme Maranhão (2015) sua relevância é maior em localidades não eletrificadas, pois em ambientes remotos, onde não existe infraestrutura disponível para a distribuição da energia elétrica ou condições para a

eletrificação, o atendimento às necessidades humanas básicas fica prejudicado, como o abastecimento de água.

O GFV possui característica intermitente no que se refere ao despacho da potência gerada e, no caso do SFB em conexão direta (que não utiliza acumuladores de energia elétrica, banco de baterias), necessita de controlador capaz de garantir a operação adequada do sistema (ALONSO ABELLA; LORENZO; CHENLO, 2003).

Nesse sentido, para os fins de compreensão e delimitação deste estudo, define-se o SFB como o sistema de conexão direta sem baterias destinado unicamente ao bombeamento. Portanto, pode-se delimitar seus principais elementos conforme se segue:

- GFV: é responsável pela transformação da energia solar em energia elétrica;
- Atuador: direcionam a energia entre os painéis solares e o conversor de frequência, de modo a proteger os elementos;
- CF: São responsáveis por transformar a tensão da fonte de energia c.c. (contínua - GFV) ou c.a. (alternada - rede elétrica) em c.a. com a modificação das características de tensão, frequência e fase;
- Carga: demanda energética a ser atendida, a motobomba.

O sistema independente que usa como fonte o GFV necessita dos equipamentos acima para garantir a continuidade do fornecimento e para ser condicionado ao aproveitamento em cargas, devido às constantes mudanças no potencial de geração de um GFV, que podem ser causadas por variações tanto na temperatura de operação dos módulos quanto nos valores de irradiância incidente.

As variações bruscas nos valores de irradiância são responsáveis por condições inadequadas na operação do sistema fotovoltaico que apresenta conexão direta com cargas motrizes, representando perda no rendimento do conjunto e até mesmo interrupção do funcionamento (BRITO, 2006).

Como dito anteriormente, sistemas para bombeamento de água, do tipo fotovoltaico (FV), possuem crescente relevância em relação ao atendimento de populações em localidades não eletrificadas. Nessas localidades, a aplicação dos sistemas FV de bombeamento com acoplamento direto pelo uso do CF, ou seja, sem acumuladores eletroquímicos, torna-se uma opção economicamente mais viável conforme apresentado por Brito (2006).

Dentro desse tema, o uso de dispositivos comerciais disponíveis no mercado nacional, tais como conversores de frequência e motobombas, tornam-se objeto de estudos para melhorar a confiabilidade e o desempenho do sistema, por meio da aplicação de técnicas avançadas de controle para garantir maior eficiência do sistema (MARANHÃO, 2015).

Para maior aprofundamento sobre o SFB são descritas a seguir as características dos equipamentos do sistema que têm parâmetros que influenciam no desempenho deste trabalho, tendo neste caso maior relevância; são eles o gerador fotovoltaico, já que o objetivo do presente estudo é o seu melhor aproveitamento; e o conversor de frequência, dispositivo que possibilitará a aplicação da técnica proposta.

2.2.1 Gerador fotovoltaico

A geração de energia elétrica por meio do GFV ocorre inicialmente nas células fotovoltaicas, que são compostas de materiais semicondutores com a devida preparação para o aproveitamento do efeito fotovoltaico, com o intuito da sua utilização para produção de energia.

O efeito fotovoltaico ocorre quando a luz incide sobre a célula fotovoltaica, de modo que o efeito fotoelétrico faz com que os semicondutores dopados com lacunas e elétrons fiquem com uma tensão entre eles e o fechamento do circuito resulta na movimentação contínua dos elétrons. Enquanto a luz continuar a incidir sobre a célula, o fluxo de elétrons se manterá, configurando uma corrente elétrica. A intensidade da corrente gerada variará proporcionalmente conforme a intensidade da luz incidente (CRESESB, 2014).

Devido à baixa potência gerada por uma célula fotovoltaica, é necessário que se associe a potência gerada, agrupando-as, formando módulos fotovoltaicos. A criação desses módulos se dá pela associação tanto em série quanto em paralelo das células, que se comportam eletronicamente como associação de baterias, dessa forma, os arranjos de células em cada tipo de módulo são realizados de modo a dar características de corrente e tensão diferenciadas de acordo com o arranjo das células (CARNEIRO, 2010). É mostrado na figura 03 a curva de tensão de circuito aberto (V_{oc}) pela corrente de curto circuito (I_{sc}) de um módulo.

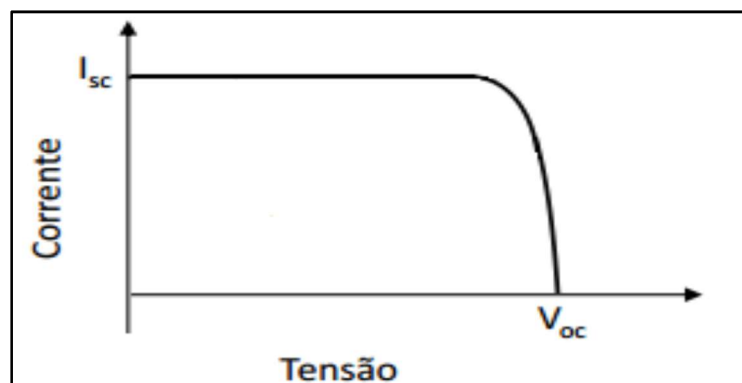


Figura 03: Curva I x V de um módulo Fotovoltaico

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

Dessa mesma forma, com a objetivo de obter maior potência de geração, são associados os módulos fotovoltaicos, que se comportando também como a associação de células, podendo serem associados em série, em paralelo e ligação mista, de modo a obter o perfil de tensão e corrente mais adequado à aplicação pretendida. Logo, o GFV é o resultado de diversas associações entre módulos fotovoltaicos.

A curva de tensão de circuito aberto (V_{oc}) pela corrente de curto circuito (I_{sc}) para as associações de módulos é mostrado na Figura 04.

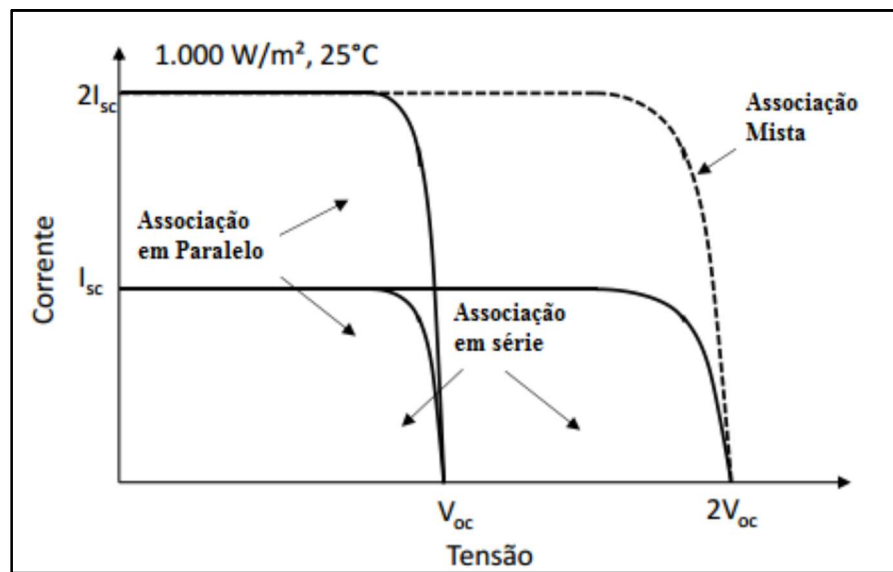


Figura 04: Curva I x V para a associação de módulos fotovoltaicos - Série, paralela e mista.
Fonte: COSTA; GUIMARÃES JUNIOR (2017)

Um ponto muito importante na associação de módulos é a atenção para módulos com potências diferentes pois, conforme mencionado, seu comportamento é similar ao de uma bateria em um circuito eletrônico, desse modo, ao introduzir um módulo de potência diferente isso pode ocasionar em perdas uma vez que tal módulo pode ser visto como carga (CARNEIRO, 2010).

Se o módulo apresentar corrente de curto-circuito menor que a corrente de operação, na associação em série, ou, se associado em paralelo, o módulo apresentar tensão de circuito aberto menor que a tensão de operação da associação, tal módulo será enxergado como carga, dissipando potência do sistema e até danificando o módulo (CARNEIRO, 2010).

Com atenção à observação anterior, é importante buscar a melhor eficiência na obtenção de energia pelo sistema fotovoltaico, com isso é dito que o ponto de melhor funcionamento desse sistema é chamado de ponto de máxima potência, e sistemas fotovoltaicos, de modo geral, devem buscar esses pontos de operação (MARANHÃO, 2015).

Tal ponto é a melhor combinação de corrente e tensão fornecidas para buscar a maior potência de geração pelo sistema, esse ponto é influenciado por condições ambientais, com relação à variação da temperatura e irradiância (VALENTE, 2011).

Conforme a Figura 05, abaixo, a influência da irradiância em uma célula fotovoltaica, se mantida a mesma temperatura, se dá pelo aumento proporcional da corrente elétrica gerada com o aumento da irradiância incidente sobre o GFV, enquanto que a tensão de circuito aberto (V_{oc}) aumenta de forma logarítmica (SANTOS; MELO, 2015).

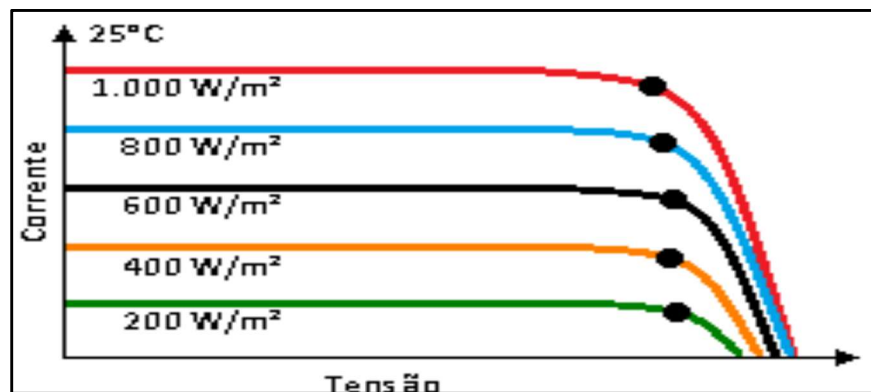


Figura 05: Influência da variação da irradiância na curva I x V de uma célula fotovoltaica de silício cristalino à temperatura de 25°C.

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

Considerando que o GFV irá trabalhar em condições que inevitavelmente elevarão sua temperatura, é importante ressaltar que o aumento de temperatura tende a reduzir a eficiência dos módulos pois, com esse aumento, a tensão de circuito aberto diminui significativamente enquanto que a corrente de curto-circuito sofre uma leve elevação (PEROZA, 2015). Este processo será, a seguir, ilustrado na Figura 06.

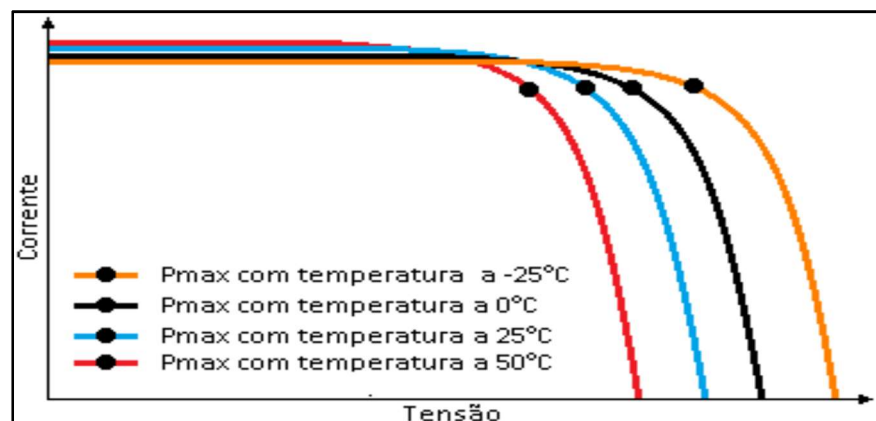


Figura 06: Efeito causado pela temperatura das células na curva I x V (para 1000w/m) em um módulo fotovoltaico

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

Verificou-se as influências da irradiância e da temperatura na busca pelo melhor aproveitamento do módulo fotovoltaico, contudo, a característica que mais interfere na variação momentânea de despacho de potência é o sombreamento parcial sobre os módulos, conforme mostrado na figura 07.

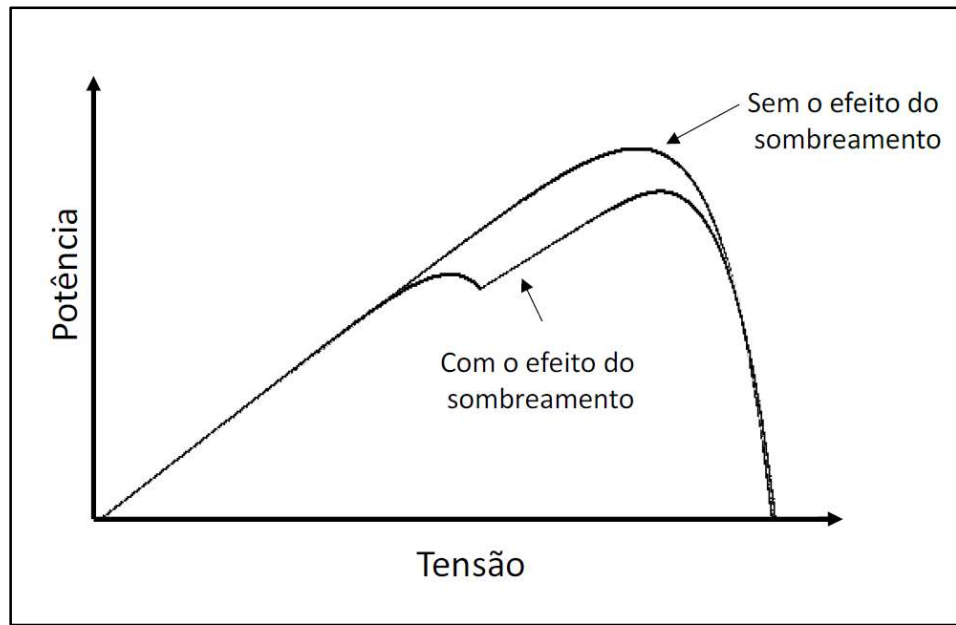


Figura 07: Efeito do sombreamento em um arranjo fotovoltaico, curva P-V

Fonte: MARANHÃO (2015)

Os efeitos do sombreamento produzem consequências em relação tanto a eficiência quanto à segurança dos módulos, este sombreamento geralmente é resultado da presença de folhas, sujidades de diversas procedências (CARNEIRO, 2010). Cabe ressaltar que no caso do aparecimento de nuvens ocorre geralmente o sombreamento total dos módulos, diminuindo de modo geral a irradiância e porventura o despacho de potência, da mesma forma como é mostrado na figura 05.

O módulo que é sujeito ao sombreamento parcial é visto como carga pelos outros módulos, com isso, é formado nessa região uma alta resistência impedindo o fluxo de corrente, caso essa corrente seja alta o suficiente para atravessar o módulo há a formação de um ponto quente (*hot spot*), elevando a temperatura de modo que possa danificar o módulo como um todo (MPPSOLAR, 2011).

Sendo assim, introduziu-se uma solução técnica, que é a inserção dos chamados diodos de *by-pass*, que desviam a corrente da célula ou módulo sombreado, evitando o aparecimento de tensões inversas nas células obscurecidas, retirando a produção dos módulos abrangidos pelo circuito do diodo, mas permitindo assim que seja mantida a produção (CARNEIRO, 2010).

Em relação a este trabalho, a atenção maior é sobre o efeito do sombreamento parcial e por sombreamento total que gera mudanças abruptas no fornecimento de energia, pois conforme é abordado no capítulo 4, ambos os casos tornam o aproveitamento da energia gerada de forma não linear menos eficiente.

2.2.2 Conversor de Frequência

A eletrônica de potência contribui amplamente para o desenvolvimento de tecnologias de manipulação de potência, em relação ao acionamento de cargas motrizes, pode-se dizer que uma das maiores contribuições é por intermédio de conversores de frequência (CF), que se encontram aprimorados a ponto de realizar a entrega de potência em c.a. com variação de parâmetros como tensão, corrente e frequência da fonte de energia (CAPELLI, 2002).

Essa capacidade é decorrente do funcionamento deste dispositivo, que geralmente transformam a forma da corrente de c.c. para c.a., possibilitando assim a variação da potência entregue a uma carga, com isso, sua utilização é mais comum no meio industrial, no qual realiza acionamento e controle de motores de indução. Em sua utilização para manipulação da potência entregue a uma carga motriz, observam-se uma série de vantagens, como operar a carga em condições diversas às suas características construtivas (DE MANEIRA, 2013).

Está difundido comercialmente uma diversidade de modelos de CF no que se refere à potência que eles suportam, metodologias de controle, tecnologia de programação e/ou parametrização, dentre outras características. Quanto à descrição do CF, será abordado o modelo convencional, que apresenta o diagrama elétrico conforme a figura 08 (MARANHÃO, 2015).

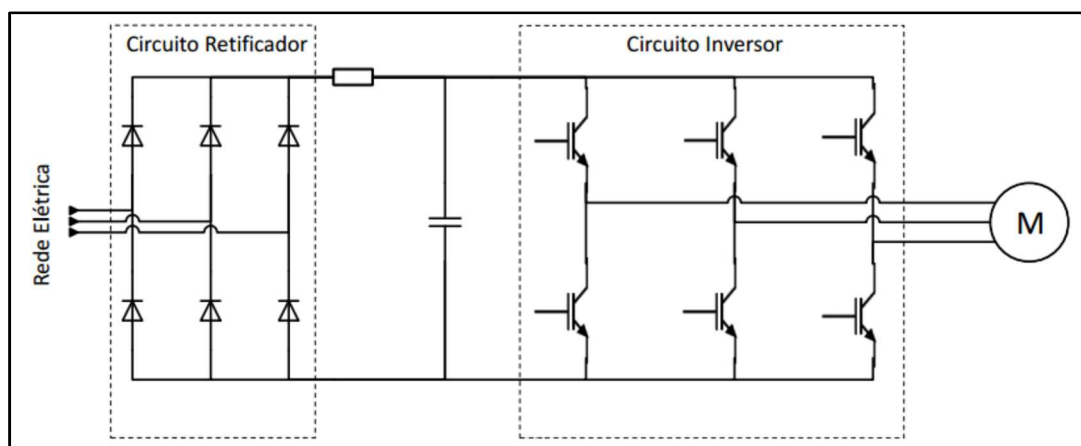


Figura 08: Diagrama elétrico básico de um conversor de frequência convencional

Fonte: MARANHÃO (2015)

Deve-se observar, na aplicação de CF em SFB, os aspectos inerentes da produção de energia de forma que o aparelho seja integrado de forma concisa à aplicação (MARANHÃO, 2015). Para a utilização de CF é de suma importância a capacidade de suporte a controles de velocidade (externo ou embarcado), sendo uma vasta variedade desses aparelhos operantes com uma base de controle de velocidades já embarcadas, sendo a mais comum o controlador PID (WEG, 2003; WEG, 2010; e WEG, 2015).

Para aparelhos que não dispõem dessa base, uma entrada configurável precisa ser disponibilizada ao usuário, proporcionando a possibilidade de conectar um sinal externo de microprocessamento do CF e permitindo a inserção de um sinal como referência do PWM criado pelo CF.

2.2.3 Motobomba

Os tipos de cargas alimentados por CF são controlados por dois métodos: vetorial e escalar. Sendo assim, podem ser utilizados os dois tipos de Motobombas disponíveis no mercado: as de deslocamento positivo, com torque constante; e as centrífugas, com torque variável. Além disso, para bombeamento de água, são utilizadas motobombas centrífugas de acordo com a altura manométrica a qual a bomba será aplicada (BRITO, 2006).

A alimentação de motores síncronos de ímãs permanentes pode ser feita por um GFV, já que aqueles podem ser conduzidos por uma ampla faixa de variação de potência de alimentação, decorrente das condições de irradiância. Também há possibilidades de um possível emprego de motores de indução para condução de uma Motobomba submersa com a potência gerada por um GFV (MARANHÃO, 2015).

Nas duas ocasiões faz-se necessário a utilização de motores trifásicos para aplicações de grandes magnitudes. Todavia, para aplicações de baixa potência são empregados motores de indução monofásicos uma vez que este produz um torque de partida elevado, indispensável para a operação da Motobomba (SHAW; GAUR, 2014).

Abordados os principais pontos para o embasamento teórico do SFB, com a descrição dos seus componentes e forma de obtenção energia, destacando os pontos mais relevantes para este trabalho, enfim segue o último ponto desta fundamentação, o controle dinâmico de sistemas.

2.3 Controle Dinâmico de sistemas

De acordo com Ogata (2002), verifica-se que para a realização do controle de um sistema é necessário a utilização de um modelo e não é possível expressar na prática o caráter aleatório e valores instantâneos, que são resultantes de perturbações e dinâmicas reais; nesse sentido, utiliza-se sinais de teste para se analisar a resposta do sistema e aproximá-la a um modelo conhecido, através da observação das respostas transitória e estacionária.

Sendo assim, a ação de um controlador é feita a partir da comparação do valor de saída do processo e o valor desejado, gerando um sinal de desvio ou erro, com isso, produzindo um sinal de controle de modo a minimizar esse erro.

Ainda com base em Ogata (2002), as ações de controladores analógicos industriais pode ser liga-desliga ou contendo combinações entre as ações proporcional, integral e derivativo, que para a realização de um controle em malha fechada, ou seja, com observância do sinal de saída, necessitam de um elemento de medição para verificação de desvio.

Por conta disso, pode-se verificar que cada ação básica de controle tem suas características de atuação e resposta, sendo abordado a seguir os controladores proporcional, integral e derivativo.

Sabendo que $u(t)$ é o sinal de saída do controlador no domínio do tempo, podemos considerar que o controlador proporcional é basicamente um amplificador com ganho ajustável de acordo com sua configuração, já que sua saída é uma variação proporcional do sinal de desvio, conforme a equação abaixo, que representa sua ação de controle no domínio do tempo:

$$u(t) = K_p e(t) \quad (2.6)$$

Por sua vez o controlador integral, também chamado de controlador de reestabelecimento, atua de modo a variar o sinal de saída segundo uma taxa proporcional ao sinal de desvio, ou seja, caso dobre o valor de desvio também é dobrado o tempo da ação de controle, e se for anulado o desvio a saída permanece estacionário. Sendo assim, segue abaixo sua ação de controle no domínio do tempo:

$$u(t) = K_i \int_0^t e(t) dt \quad (2.7)$$

O controlador derivativo, também chamado de controle de taxa, tem o efeito de modificar a amplitude da saída do controlador de modo proporcional ao sinal de erro, dessa forma, agindo de modo a antecipar o erro atenuante e a ação corretiva necessária, resultando no aumento da estabilidade do sistema, contudo amplificando os sinais de ruídos e perturbações, podendo causar um efeito de saturação no atuador.

Diante do exposto verifica-se que a ação derivativa só pode ser utilizada em conjunto com outro controlador pois sua ação é apenas efetiva durante o efeito transitório. Por conta disso, no controlador proporcional derivativo, de alta sensibilidade, sua ação é associada ao controlador proporcional, conforme mostrado na sua equação de ação no domínio do tempo:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} \quad (2.8)$$

Por fim, seguindo o estudo de Ogata (2002), verifica-se que cada associação das ações de controle descritas acima resulta em um controlador com características próprias, resultantes da combinação das características das ações de controle.

Nesse sentido, é associado o controlador proporcional-e-integral, proporcional-e-derivativo ou os três, proporcional-integral-derivativo, gerando o controlador PID, que combina a vantagem de cada uma das três ações de controle, conforme sua resposta no domínio do tempo:

$$u(t) = K_p e(t) + K_p T_d \frac{de(t)}{dt} + \frac{K_i}{T_i} \int_0^t e(t) dt \quad (2.9)$$

Ocorre que as ações de controle clássico, apresentadas acima, agem pela adaptação dos ganhos relacionados aos controladores à modelagem do sistema, sendo robustos na medida que o sistema se comporta de acordo com o esperado. Tal comportamento é devido ao fato dos referidos ganhos serem estáticos, ou seja, o controle do sistema dinâmico é realizado por parâmetros pré-estabelecidos.

Por conta disso, ocorre o desenvolvimento de metodologias e técnicas que são capazes de modificar seus parâmetros na medida em que as características do sistema controlado são modificadas, nesse sentido, é apresentado a seguir noções sobre o controle adaptativo, a fim de apresentar uma nova abordagem para suprir necessidade de sistemas que possuem modelos de parâmetros variáveis.

2.3.1 Controle adaptativo de sistemas dinâmicos

Com o desenvolvimento do estudo sobre controle, foram desenvolvidas metodologias diversas ao controle analógico clássico, com o intuito de obter respostas com características mais robustas à sistemas não lineares. Nesse sentido, verifica-se o uso de controle adaptativo, através de heurísticas ou inteligência artificial como métodos para controle de sistemas, ou até mesmo a combinação de técnicas, como se observa pelo trabalho de Al-Dunainawi, Abbod e Jizany (2017).

Observa-se em Ogata (2002) que a teoria de controle clássico pode resolver sistemas não lineares pelo uso do método de linearização. Contudo, existem casos em que as características operacionais fixas desses controladores não se adaptam à sistemas com variações nos processos envolvidos na dinâmica de controle.

Sendo assim, pode-se definir o controle adaptativo como sendo aquele que é capaz de ser aplicado a sistemas cujas variáveis evoluem conforme seu próprio comportamento, como também é capaz de estabilizar tais variáveis e impor trajetórias de referência. Portanto, o controle adaptativo é aquele capaz de modificar seus parâmetros de acordo com o comportamento do sistema. A utilização da metodologia de controle mencionada é apresentada como uma alternativa para o controle convencional, pois a maioria dos sistemas reais possuem características não lineares, com variações de parâmetros, atraso de transporte e perturbações diversas (ROCHA et al, 2012).

Dentre as vantagens da utilização de controles adaptativos está a identificação dos sistemas, pois em alguns casos o conhecimento detalhado de um sistema não linear pode ser difícil e oneroso. Nesse sentido, os controles adaptativos permitem a identificação dos sistemas por meio do uso de modelagens empíricas, de dinâmica linear aproximada ou de ordem pré-estabelecida (ROCHA et al, 2012).

Observa-se que paradigmas de busca em geral podem ser utilizados para o controle adaptativo, sendo que seus desempenhos variam conforme as características de cada um. No estudo realizado por Al-Dunainawi, Abbod e Jizany (2017), é mostrado um controlador ANFIS (*adaptive neuro-fuzzy inference system*), no qual é utilizado o PSO para auxílio na modelagem do sistema e busca dos parâmetros do controlador ANFIS.

Com base no que foi abordado é tratado a seguir sobre o DPCM, que serve ao fim de ser elo entre técnicas de controle e sistemas em que atua CF recebendo do DPCM e aplicando o sinal de controle à carga.

2.3.2 Dispositivo de Programação de Controladores e Monitoramento

O DPCM foi desenvolvido com o objetivo de se conectar ao CF através de entrada auxiliar. Sendo produzidas as placas com dois microcontroladores Atmega 328 pertencentes à fabricante ATMEL compatível (pelo uso de circuitos simplificados de condicionamento de sinais) com sistemas fotovoltaicos no que concerne a conexões com CF e alimentação de circuitos, de modo que são compatíveis ou se adaptam aos valores de entrada e saída de corrente e tensão dos dispositivos utilizados (COSTA; GUIMARÃES JUNIOR, 2017).

Em seu desenvolvimento se optou por utilizar dois núcleos de processamento, tanto para sistema de controle, quanto para monitoramento do sistema. Esse tipo de fracionamento de processos permite um maior número de portas analógicas para a obtenção de dados, garantindo também uma maior velocidade para processamento de dados e assim aumentar a rapidez de resposta para o controle. Dessa forma, a programação do dispositivo é realizada com base na linguagem computacional C++, relacionada a plataforma de programação. O controle dos núcleos de processamento Mestre e Escravo utilizou uma lógica de programação representada nos fluxogramas das figuras 09 e 10 (COSTA; GUIMARÃES JUNIOR, 2017).

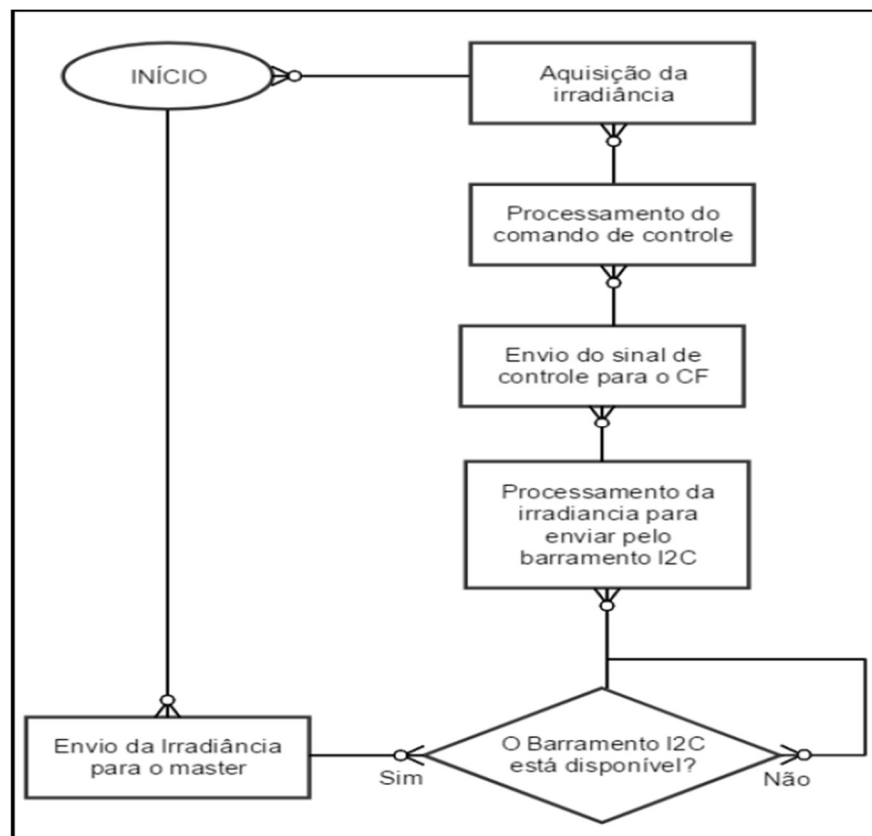


Figura 09: Fluxograma da lógica de controle do microcontrolador Escravo

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

A programação do controlador Escravo é a responsável por enviar o sinal de controle para o CF, concebido para modificar a velocidade da motobomba. O código é iniciado com a coleta dos dados de irradiância; em seguida o microcontrolador dará início ao processamento do controle enviando o comando para o CF. A Figura 09 representou o fluxograma do esquema de programação do microcontrolador Escravo. Para finalizar, o controlador envia para o Mestre através do barramento I2C as informações de irradiância coletadas pelo transdutor para que este possa fazer o armazenamento dos dados. Ao final deste processo a programação é reiniciada (COSTA; GUIMARÃES JUNIOR, 2017).

O código do Mestre é responsável pelo sistema de aquisição e armazenamento de dados. A programação inicia com a aquisição dos dados de irradiância transmitida pelo Escravo, em seguida é feita a aquisição dos dados dos demais transdutores. Por fim, o barramento I2C é novamente utilizado para captura os dados de data e hora e o processo de armazenamento das informações é efetuado. Ao final deste ciclo a programação reinicia. A figura 10 apresenta o fluxograma do controlador Mestre (COSTA; GUIMARÃES JUNIOR, 2017).



Figura 10: Fluxograma da lógica de controle do microcontrolador Mestre.

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

Sendo assim, verifica-se pelo trabalho de Costa e Guimarães Junior (2017) que este dispositivo permite a programação livre baseada na IDE (*Integrated Development Environment*) do Arduino, o que permite a implementação de qualquer técnica de controle adaptativo com algoritmo que possa ser traduzido naquela IDE. Além disso, foi concebido associado aos transdutores de irradiância, tensão, corrente e equipamento auxiliar para registro de dados, com isso, tornando-se um equipamento bastante útil para uso em controle.

Com as informações acima se encerra a fundamentação teórica deste estudo, com isso, segue o capítulo 3 com os apontamentos de como este trabalho foi conduzido, com as devidas abordagens de todos os principais pontos sobre técnicas, equipamentos e considerações utilizadas para realização da proposta.

3 MATERIAIS E MÉTODOS

Neste capítulo será descrita a proposta de controle no sistema, ou seja, o apontamento do uso do PSO como técnica de controle gravada no DPCM, que por sua vez aplica o sinal de controle em entrada auxiliar do CF, que controla a frequência da tensão de alimentação da motobomba da bancada de testes de SFB de modo a melhor aproveitar o despacho de potência intermitente característico desse sistema.

Será especificado a forma de obtenção do sinal de controle aplicado, que é o valor de posição das partículas, esse age como um sinal proporcional adaptativo, que é calculado através do algoritmo do PSO na busca pelo menor valor de erro entre o sinal de tensão no barramento c.c. do GFV e o *setpoint* desejado.

Dessa forma, o presente capítulo apresenta, no item 3.1, a bancada de ensaios, a qual possui os dispositivos básicos de um SFB; o DPCM no subitem 3.1.1, elemento integrador dos dispositivos de monitoramento e responsável por abrigar o algoritmo de controle e aplica-lo ao CF; finalizando a descrição dos materiais utilizados.

Quanto à descrição metodológica, é apresentado no item 3.2 a metodologia de controlador proporcional PSO com variação adaptativa do sinal de controle, sendo descrito e comentado o algoritmo básico de controle no subitem 3.2.1, finalizando a apresentação da metodologia.

No Apêndice C deste trabalho é mostrado a simulação do controlador PSO proporcional em motor c.c., a qual tem o intuito de apresentar técnica similar à que está sendo proposta, mas com uso das partículas como ganho K_p de um controlador proporcional, esta foi realizada antes da proposta deste trabalho para analisar o comportamento do PSO e sua viabilidade para controle.

3.1 Bancada de Ensaios de SFB

A realização dos testes da proposta deste trabalho será realizada através da operação da bancada do Laboratório de Automação e Controle presente na Universidade Federal do Amapá – UNIFAP, esta é representada esquematicamente na Figura 11.

Essa fora reproduzida baseada em uma bancada inicialmente concebida pelo Laboratório de Sistemas Fotovoltaicos da Universidade de São Paulo, para realização controle de altura manométrica em testes experimentais, e logo após replicada pela UFPA (MARANHÃO, 2015).

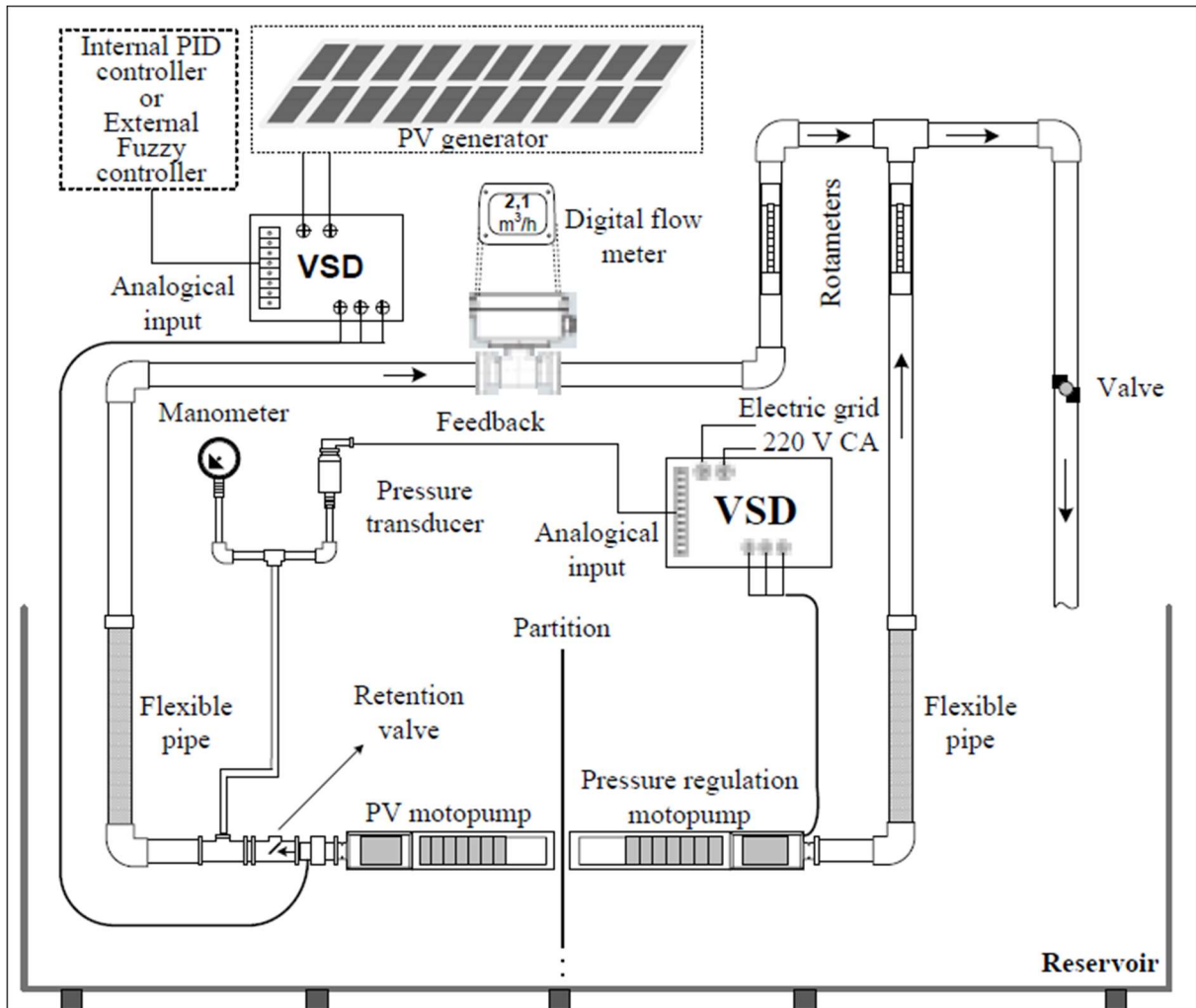


Figura 11: Diagrama esquemático da bancada de ensaios

Fonte: Modificado de MARANHÃO (2016)

Ocorre que a bancada de ensaios de SFB foi utilizada de forma diversa a sua destinação, pois, em suma, existia uma divergência entre as bancadas originalmente construídas e a do laboratório de automação e controle da UNIFAP, já que esta última estava com o transdutor de pressão defeituoso durante a realização deste trabalho, o que impossibilitou o simulador de valores de altura manométrica.

Por conta disso, durante os testes não foi realizada a simulação de altura manométrica, mas tão somente o uso do GFV para alimentação do sistema que utiliza o CF para receber o controle e aplica-lo à motobomba, ou seja, a bancada de ensaios em si não foi utilizada, mas sim apenas alguns de seus componentes e fora dos parâmetros ideais de operação, já que a motobomba, por sua vez, opera sem nenhuma aplicação de pressão correspondente a sua pressão manométrica de operação e, por conta disso, opera em condições diferentes da sua faixa de operação comercial.

A Figura 12 mostra a visão aérea dos módulos fotovoltaicos da bancada que dispõe do arranjo do sistema motobomba-CF alimentado por este GFV, em que são executados os testes de controle e velocidade mediante controle externo ao CF. Arranjo este que é composto por gerador estruturado por 19 módulos de 55wp de silício amorfo, postos em 4 linhas em paralelo, com 5 módulos em série de cada uma, excluindo a linha final, com 4 módulos em série, não fazendo parte do GFV o módulo destacado na imagem.



Figura 12: Foto dos módulos fotovoltaicos em cima do Prédio do Bloco de Engenharia Elétrica.
Fonte: COSTA; GUIMARÃES JUNIOR (2017)

O sistema conta com duas motobombas submersas do tipo centrífuga, sendo uma de 8 estágios possuindo 0,5 cv (não utilizada nos testes) e outra de 2 cv com 13 estágios (ligada ao GFV), o CF utilizado é de 2 cv do fabricante WEG, modelo CFW10. O sistema elaborado pelo laboratório da Universidade federal do Amapá é mostrado na figura 29. Abaixo seguem as tabelas 01 e 02, com as informações técnicas sobre os equipamentos presentes na bancada.

Tabela 01: Características físicas e elétricas da motobomba em teste

Potência Nominal	2cv (1,47kW)
Velocidade	3450 RPM
Tensão (Ligada a rede Elétrica)	220V
Corrente Máxima (Imax)	9,7A
Frequência	60Hz

Fonte: Elaborado pelo autor

Tabela 02: Características do Módulo Fotovoltaico utilizado

Potência Nominal (P_{mp})	55Wp
Tensão Nominal (V_{mp})	17,83V
Corrente Nominal (I_{mp})	3,08A
Tensão Circuito Aberto (V_{oc})	22,07V
Corrente de Curto-Circuito (I_{sc})	3,28A
Área	0,416m ²

Fonte: COSTA; GUIMARÃES JUNIOR (2017)

O sistema real da bancada descrita está devidamente apresentado na Figura 13.

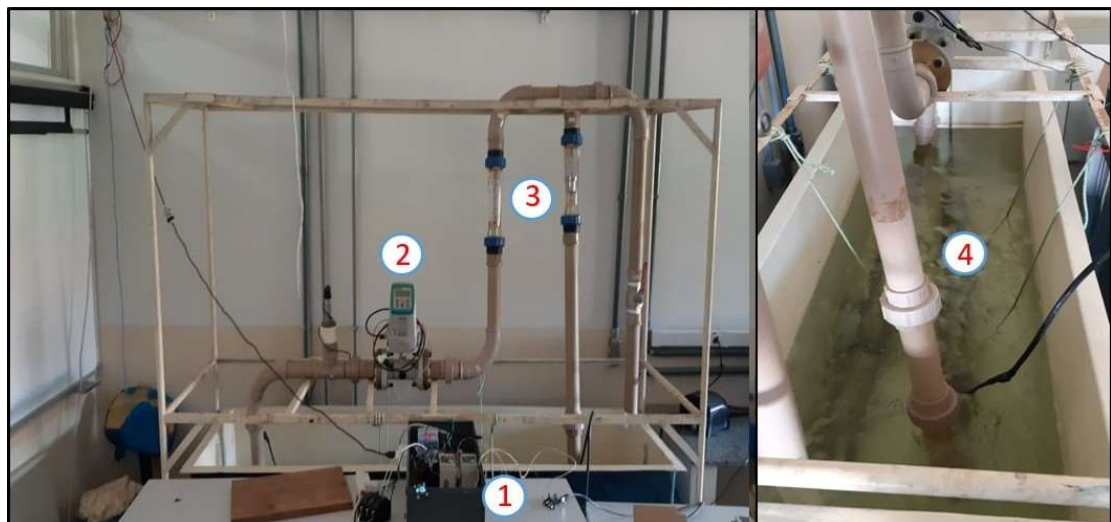


Figura 13: Foto da bancada de ensaios de SFB da UNIFAP com a identificação dos seus componentes

Fonte: Elaborado pelo autor

- 1 – Conversor de frequência;
- 2 – Transdutor de vazão (Com transmissor acoplado);
- 3 – Rotâmetros (Não possuem função nos testes);
- 4 – Motobomba.

Segundo o fabricante WEG (2010), o CF utiliza a tensão da rede elétrica de amplitude e frequência fixas para gerar tensão de amplitude e frequência variáveis, por PWM, para o controle da velocidade de motores de indução, sendo assim, para a aplicação na bancada de ensaios a alimentação é realizada no seu barramento c.c. com a tensão do GFV utilizando seu circuito inversor alimentar a motobomba com tensão c.a., considerando que para cada tipo de

motobomba os CFs devem possuir sistema de ajuste da amplitude e da frequência fundamental da tensão gerada para que possa ser gerado o torque adequado à motobomba utilizada.

Ainda com relação aos CFs, é apresentado por Maranhão (2015) que a parametrização desejada pode ser realizada pelas suas entradas e/ou saídas, analógicas ou digitais, que podem ser habilitadas ou desabilitadas utilizando-se a interface homem máquina, obedecendo aos índices dos parâmetros relacionados. Aproveitando esta característica o DPCM, descrito no subitem 2.3.2, foi desenvolvido e aplicado na bancada de ensaios utilizada neste trabalho.

3.1.1 DPCM aplicado à bancada de ensaios

Para a correta e eficaz aplicação do método proposto, o DPCM é um dos principais componentes, pois através dele será aplicada a técnica do PSO na bancada de ensaios. Seu funcionamento se dá pela ação de microcontroladores programáveis com a IDE do Arduino, sendo que a saída do sinal de controle é aplicada ao CF da bancada de ensaios do SFB.

Conforme se verifica pelo proposto por Costa e Guimarães Junior (2017), a estruturação dos fluxogramas de programação dos microcontroladores do DPCM, mostradas no subitem 2.3.2, não leva em consideração a lógica *Fuzzy*, método de controle utilizado durante o desenvolvimento do DPCM. Por conta disso, observa-se que a programação do microcontrolador mestre independe do controle aplicado, pois este é responsável pelo monitoramento dos transdutores, logo, as alterações serão realizadas apenas no microcontrolador Escravo, que recebe o método de controle.

No caso deste trabalho, a informação dos transdutores será utilizada unicamente para fins de registro, pois já que é proposto o controle de *setpoint* fixo de tensão, o único desses dispositivos que irá interferir na técnica é o transdutor de tensão, responsável pelo monitoramento da variável de controle.

Sendo assim, não haverá mudança no fluxograma de programação do DPCM, a modificação na metodologia de controle se dá pelo uso apenas do transdutor de tensão para o controle, e não mais a irradiância. Com isso, implica-se em realizar alterações pontuais no método de programação do DPCM, de modo a melhor aproveitá-lo para a técnica proposta.

Por conta disso, é utilizada a saída do transdutor de tensão tanto no microcontrolador Mestre, ao qual é originalmente destinado; quanto no Escravo, para que este possa receber o sinal de saída a ser controlado. Ademais, não houveram alterações no DPCM, e o único microcontrolador a ter sua programação alterada é o Escravo, do qual foram substituídas as rotinas de controle.

São apresentados nas tabelas 3 e 4, a ficha técnica do DPCM e a característica dos equipamentos do sistema de medição, respectivamente.

Tabela 03: Ficha técnica do DPCM

Tensão de alimentação (GFV)	126 – 548V
Tensão Auxiliar de alimentação	8 – 32V
Tensão de entrada dos sensores	0 – 5V
Pinos de entrada Analógica	0 – 40mA
Tensão de saída do Controlador	0 – 5V

Fonte: COSTA; GUIMARÃES JUNIOR (2017), p. 37

Tabela 04: Resumo das características técnicas dos transdutores do sistema de monitoramento

Variável monitorada	Tipo de Transdutor	Faixa de medição
Irradiância	Célula Fotovoltaica Calibrada	76,53mV ↔ 1000W/m ²
Tensão na entrada	Transdutor de Tensão	0 – 500V ↔ 0 – 10V
Corrente na entrada	Transdutor de Corrente	0 – 10A ↔ 0 – 10V
Temperatura	Transdutor de temperatura	-55 a 150°C ↔ 10mV/°C
Vazão na tubulação	Transdutor de Vazão	0 – 32m ³ /h ↔ 4 – 20mA

Fonte: COSTA; GUIMARÃES JUNIOR (2017) p. 38

A gravação dos dados relacionados aos componentes de aquisição de dados é realizada em um cartão SD, do qual é obtido os dados brutos sobre a aplicação da técnica de controle. Portanto, esse dispositivo tem grande importância para a metodologia uma vez que possibilita tanto a aplicação de quaisquer técnicas de controle aceitas pelos microcontroladores, neste caso o PSO; como também a gravação dos dados operacionais.

3.2 Implementação do controlador PSO no SFB

Para o correto detalhamento metodológico deste estudo, faz-se necessárias considerações a respeito da aplicação do PSO em problemas concretos, pois quando se trata de problemas de engenharia, existe uma série de considerações a serem feitas que desconstruem o estado de sofisticação teórico das soluções de problemas puramente computacionais.

De acordo com a proposta já descrita, a busca pelo valor ideal de sinal de controle assumido pelo controlador proporcional é realizada conforme cada avaliação do sistema, obtenção do *fitness*; este é o recurso que dá dinâmica ao controle e o diferencia do uso de um controlador proporcional de ganho estático e de qualquer método de controle com baixa diversificação, que não se adapta às variações das características do sistema. Tal característica é importante pois, conforme apresentado por Maranhão (2015) pela figura 7 no subitem 2.2.1, o sombreamento parcial em um arranjo fotovoltaico provoca mínimos locais de convergência para o valor ótimo, contudo, o principal problema de convergência é o sombreamento total, que pode provocar afundamentos de irradiância e tensão, conseqüentemente.

Serão apresentadas as principais alterações e abordagens do PSO, conforme forem realizados os testes, para que se apresente a proposta que permita o uso do algoritmo neste sistema. Essa proposta gera uma abordagem de utilização direta do PSO para análise de seu comportamento sobre variações dinâmicas; porém, diferente dos estudos como o de Teo et al (2013), que utiliza o PSO como método coadjuvante no controle.

Para tanto é realizado o controle de tensão por *setpoint* fixo, pois conforme apresentado por Brito (2006), essa é a estratégia de controle mais simples, porém, não o ideal já que o sistema opera fora do ponto de máxima potência, foi utilizado o valor de 320V de setpoint por avaliação empírica, pois este valor mostrava a menor quantidade de interrupções por subtensão. Sendo assim, as partículas são utilizadas como valores de sinal de controle, minimizando o valor de erro visto pelo controlador com a alteração da frequência de funcionamento da motobomba. Sendo assim, a metodologia descrita neste capítulo está representada pela figura 14, que apresenta o diagrama elétrico desta proposta.

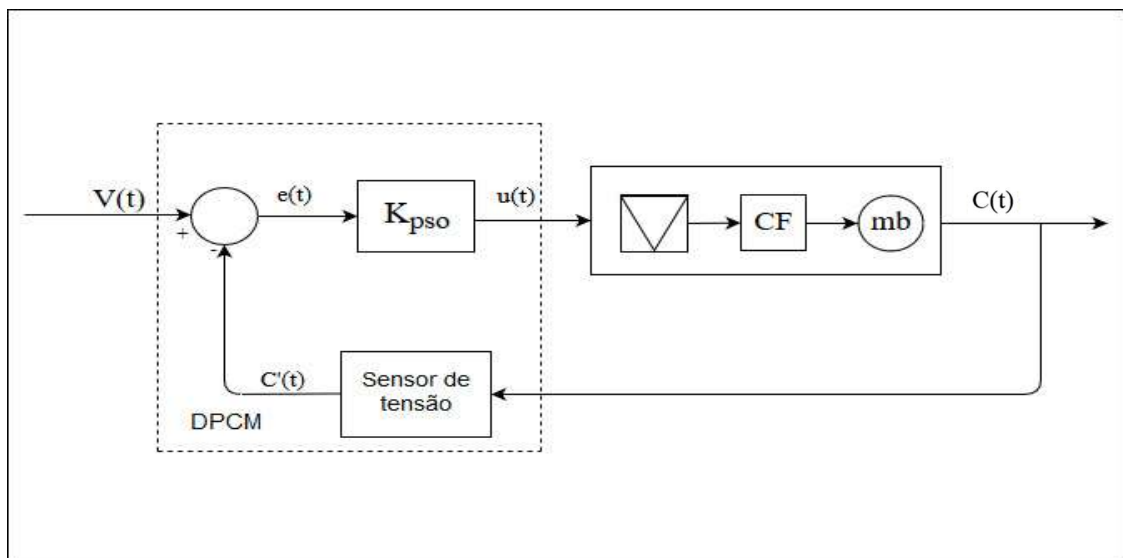


Figura 14: Diagrama elétrico simplificado do sistema de controle

Fonte: Elaborado pelo autor

$u(t)$ – ação de controle fornecida pelo DPCM;

$C(t)$ – Tensão no barramento c.c. do GFV;

$C'(t)$ – tensão no barramento c.c. do GFV entregue pelo transdutor ao DPCM;

$V(t)$ – *Setpoint* de tensão a ser mantido pela ação de controle; e

$e(t)$ – erro entre *setpoint* e tensão lida pelo transdutor, a ser minimizada pelo PSO.

Dessa forma, utilizando a característica de otimização do PSO, a busca de valores ideais de partículas, estas são aplicadas como sinal de controle proporcional, onde a posição de cada partícula corresponde a um sinal PWM que é entregue a um circuito resistor-capacitor (RC) para filtrar o sinal, transformando-o em um sinal de tensão c.c. (0 a 5V) que corresponde a $u(t)$, saída do DPCM que é entregue ao CF. Por fim, o CF entrega a tensão PWM senoidal de alimentação da motobomba, que tem sua frequência alterada de acordo com o controle aplicado.

Dessa forma, as partículas modificam sua posição conforme o algoritmo do PSO de modo a diminuir o erro, $e(t)$, entre o *setpoint* e o sinal de tensão do barramento c.c. do GFV, Sendo assim, cumpre-se o objetivo geral deste estudo de fechar a malha de controle, ou seja, utilizar a atuação provida pela técnica de acordo com valores retroalimentados do sinal de saída.

Com isso, é abordado a seguir a forma de implementação do algoritmo do PSO para fins de controle adaptativo com a descrição do pseudocódigo utilizado pelo método e a definição de cada operação do algoritmo.

3.2.1 Código do controlador PSO

É importante tratar de forma detalhada de como é aplicado o PSO como o método de controle, para tanto, se faz necessário realizar considerações sobre as diversas modificações, incrementos e parametrização realizadas no código apresentado no Apêndice B, dessa forma, com o intuito abordar essa metodologia, trata-se de cada ponto com base no pseudocódigo apresentado no referido apêndice.

Primeiramente, cabe ressaltar que o algoritmo proposto foi produzido na IDE do Arduino, que tem sua linguagem baseada em C++, por conta disso a primeira parte do código, das linhas 1 a 12, é realizada a inicialização dos parâmetros do algoritmo, abaixo segue o trecho do pseudocódigo com explicação de cada variável.

```

1  atraso ← Tempo entre aplicação do controle e leitura da tensão
2  setpoint ← valor desejado de tensão 320
3  C ← Coeficiente cognitivo
4  S ← Coeficiente social
5  fi ← C + S Fator de restrição e  $\varphi$ , descritos na equação 2.4
6  K ←  $2 / |2 - fi - (fi^2 - 4 * fi)^{0.5}|$ 

```



```

7  w ← 0.9 Constante de inércia
8  N ← 3   n° de partículas
9  aux_atualizar ← 0 contador de auxílio ao espalhamento
10 atualizar ← 2 limite para espalhamento e atualização
11 sinal_controle ← 3 Pino de aplicação do sinal de controle
12 volt_in ← A0 Pino que recebe o sensor de tensão

```

Em seguida, a primeira rotina realizada, de *setup* - linhas 13 a 19, apresenta a inicialização dos vetores das partículas, suas velocidades, melhores locais (recebem o valor da própria partícula) e avaliações (valores nulos), como se trata de inicialização, os valores assumidos não devem interferir na convergência das partículas, por isso a escolha dos valores apresentados.

```

13 rotina inicial do microcontrolador(setup)
14   para i ← 0 até N faça Inicialização das partículas
15     x[i] ← 90 + (randômico entre 1 e 10)
16     v[i] ← (randômico entre 1 e 10)
17     xlocal[i] ← x[i]
18     flocal[i] ← 0
19 fim da rotina inicial
20

```

A partir disso é dado início ao laço que será executado enquanto o sistema permanecer ativo, ou seja, enquanto não houver nenhuma interrupção no controle, como erro por subtensão. Sendo assim, o controle se inicia com um laço *for*, que realizará toda a rotina de controle para todas as partículas, tal rotina se inicia pelo cálculo de posição e velocidade das partícula, linhas 23 a 26; logo em seguida é verificado se não houve fuga das partículas ou de suas velocidades do alcance do sinal PWM (de 0 a 255), com o retorno da partícula, para valor entre 80% e 90% do valor de melhor global; e respectiva velocidade, para valor entre 70% e 130% do seu valor atual, das linhas 28 a 31.

```

21 início do código em laço
22 para i ← 0 até N faça
23   local ← C*(randômico entre 0 e 1)*(xlocal[i] - x[i])
24   global ← S*(randômico entre 0 e 1)*(xglobal - x[i])
25   v[i+1] ← K * (w * v[i] + local + global)
26   x[i+1] ← x[i] + v[i+1]
27   limitação de x nos valores de PWM entre 0 e 255
28   se x[i] > 255 ou x[i] < 0 ou v[i] > 255 faça
29     x[i] ← randômico entre xglobal-5 e xglobal+5
30     v[i] ← randômico entre v[i] * 0.7 e v[i] * 1.3
31 fim se

```

Após tais rotinas é assegurado que as partículas estão no alcance do sinal PWM, então é dado início à aplicação do sinal de controle com espera da atuação do sistema e leitura do sinal de tensão, das linhas 32 a 37.

```

32  aplicar na porta sinal_controle o sinal PWM x[i]
33      esperar atraso
34  VT ← 0
35  para j ← 0 até 50 faça
36      VT recebe e incrementa o valor lido da porta volt_in
37  VT ← VT / 50 recebe a média de 50 valores instantâneos

```

Em seguida é verificado se o sinal de tensão lido é inferior a 290 V, ou seja, 30 V a menos que o sinal de *setpoint*, caso a condição seja satisfeita a partícula assume valores randômicos entre 80% e 90% do valor do melhor global e o programa é direcionado para o rótulo **avaliar** da função *goto*, essa rotina, realizada entre as linhas 38 a 43, ocorre para que seja evitado afundamentos de tensão, resultando em erros de subtensão.

```

38  se VT < 290 faça Contra o afundamento de tensão
39      para j ← 0 até N faça
40          x[j] ← randômico entre xglobal*0.8 e xglobal*0.9
41
42      vá para a posição avaliar
43  fim se

```

Caso a condição acima não for verificada o código segue com a atualização dos melhores locais e globais, das linhas 44 a 53, que são atualizados se o erro entre o sinal de saída e o respectivo valor local ou global for menor que o erro associado ao valor armazenado para a partícula.

```

44  erro ← |VT - setpoint|
45  se erro <= |fglobal - setpoint| faça
46      xglobal ← x[i]
47      fglobal ← VT
48  fim se
49
50  se erro <= |flocal[i] - setpoint| faça
51      xlocal[i] ← x[i]
52      flocal[i] ← VT
53  fim se

```

Com isso se encerra o laço *for* de uma partícula, sendo executado este laço até a quantidade de partículas utilizadas como parâmetro e com isso o código segue até seu fim e antes de ser reiniciado o laço principal o contador “aux_atualizar” é incrementado.

Após o encerramento do laço descrito no parágrafo anterior, se o contador citador for igual ou maior que o parâmetro “atualizar” é dado início à rotina de espalhamento e atualização dos valores de referência melhor global e local.

Este trecho do código é de suma importância para que as partículas possam convergir para o valor real de *setpoint*, pois como os parâmetros do sistema mudam constantemente é necessário que se atualizem os valores de referência pois não existe essa previsão no método originalmente proposto.

A rotina de espelhamento ocorre das linhas 56 a 99 e começam pela separação do espalhamento, ou seja, se o valor de erro for menor que 20 V, então o espalhamento se dará pelo valor aleatório entre 2 a menos e 1 a mais que o melhor global, senão há uma margem maior para espalhamento, então se dará entre 2 a menos e 5 a mais que o melhor global.

```

56 se aux_atualizar >= atualizar faça
57
58     se VT - setpoint < 20 faça
59         para j ← 0 até N faça
60             x[j] ← randômico entre xglobal-2 e xglobal+1
61     Senão faça
62         para j ← 0 até N faça
63             x[j] ← randômico entre xglobal-2 e xglobal+5
64     fim se

```

A partir disso é posto o rótulo **avaliar**, na linha 66, e depois é zerado o valor da melhor avaliação global e aplicados os novos valores de partícula, nova leituras de tensão, nova verificação contra subtensão e atualização dos melhores local e global e sendo recalculada nova posição e velocidade para a partícula, zerando valor de “aux_atualizar” e retornando para as rotinas do laço principal.

Para a compreensão da forma como foi utilizado o PSO neste trabalho, segue na forma de pseudocódigo a descrição de cada trecho do código apresentado no Apêndice B, com o comentário sobre as variáveis e rotinas, aplicadas no algoritmo utilizado, na cor azul.

Com a devida apresentação do funcionamento do algoritmo do controlador PSO por meio do pseudocódigo, se encerra a apresentação da metodologia deste estudo:

4 RESULTADOS

Neste capítulo são apresentados os principais resultados obtidos no estudo desta aplicação do PSO como método de controle, bem como o cumprimento do objetivo geral deste trabalho, ou seja, a aplicação do controlador PSO no SFB; e cumprimento do objetivo específico de analisar o comportamento do controlador no referido sistema.

Os dados apresentados correspondem à aplicação do algoritmo do Apêndice B no DPCM da bancada de ensaios de SFB, no dia 04/06/2019, na utilização de meio ciclo de medição, das 12:26:05 às 17:51:05 horas, abrangendo a faixa de irradiância mostradas acima de 1.000 W/m^2 , no caso em teste foi possível verificar de forma bastante abrangente o comportamento do método pois além de haver uma ampla variação de irradiância nesta ocasião, ainda houve muitas variações meteorológicas, com sombreamento total pelo aparecimento de nuvens, o que possibilitou a obtenção de resultados para uma grande variedade de modificações no sistema.

A figura 15 apresenta o gráfico de irradiância e potência, verifica-se pela análise de potência que o controlador busca o sinal de potência acompanhando o comportamento da irradiância, havendo drástica queda de potência para valores de irradiância abaixo de 600 W/m^2 , contudo, para valores superiores de irradiância o aproveitamento da potência chegou a valores de até $1,107 \text{ KW}$ às 12:32 horas e $1,105 \text{ KW}$ às 14:20 horas, conforme destacado.

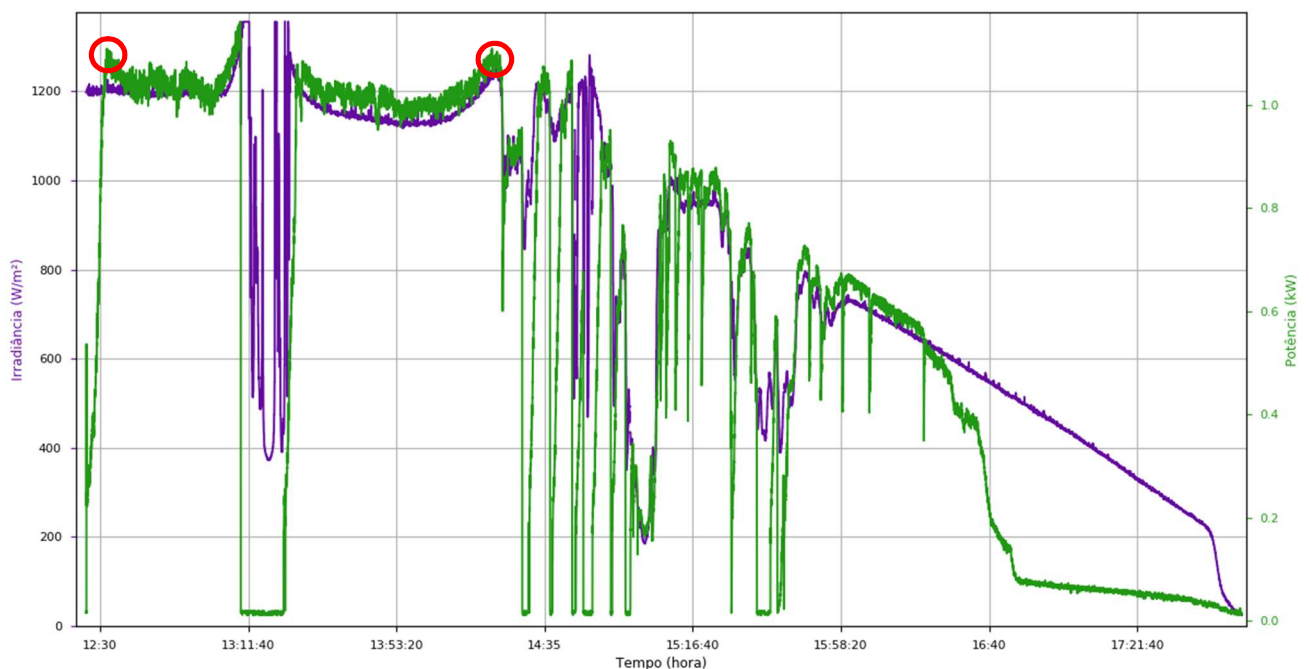


Figura 15: Irradiância x Potência entregue à motobomba com controlador PSO

Fonte: Elaborado pelo autor

A figura 16 mostra o sinal de corrente entregue à motobomba pela ação do controlador PSO, nela é possível verificar que a corrente demandada para se manter a tensão fixa é uma curva que acompanha o comportamento da potência.

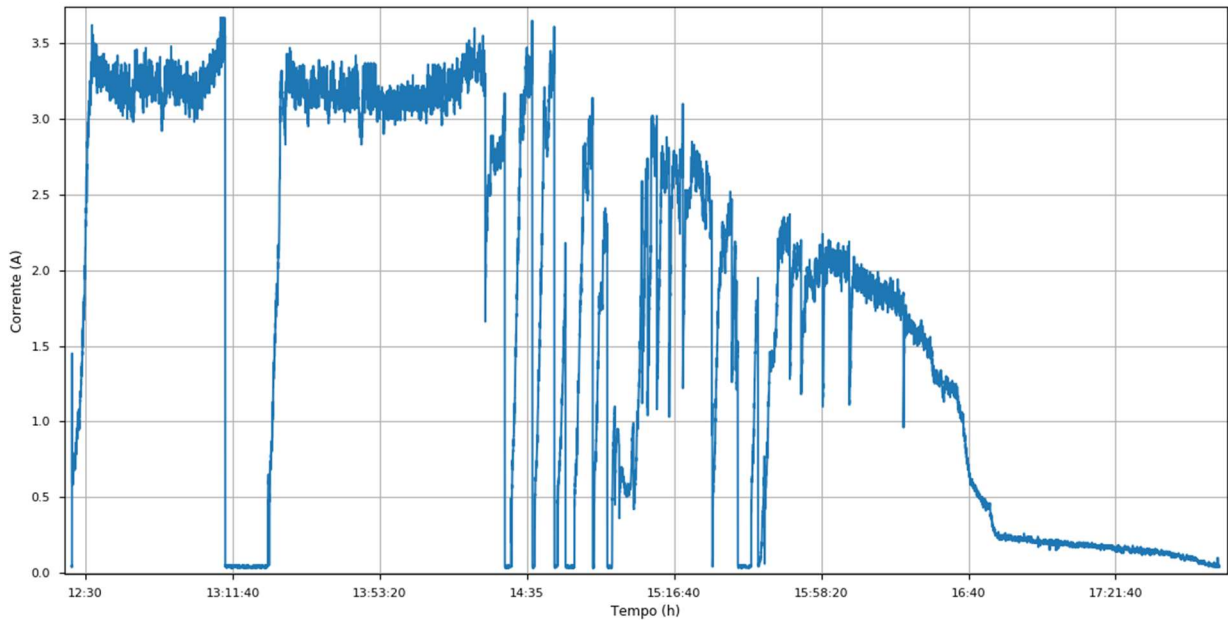


Figura 16: Corrente entregue pelo CF à motobomba

Fonte: Elaborado pelo autor

É importante analisar o perfil de tensão no barramento c.c. do CF mostrado pela figura 17, já que este é o parâmetro de controle do método, foi utilizado o *setpoint* de 320 V, destacado em vermelho.

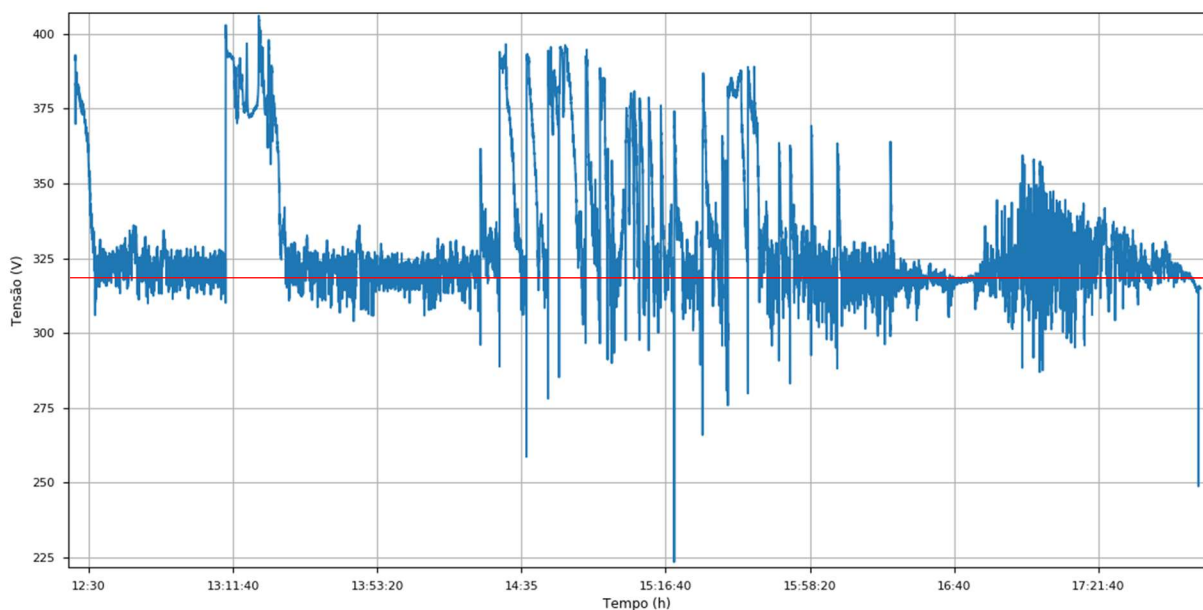


Figura 17: Tensão no barramento c.c. do CF com a atuação do controlador PSO

Fonte: Elaborado pelo autor

Com base na figura 18, verifica-se que para as faixas de irradiância superiores à 1000 W/m^2 o controle apresentou resultados convergentes que orbitam o valor desejado, gerando diferenças de tensão com amplitudes de até 15 V para mais e para menos do valor desejado, isso se dá pela própria natureza exploratória do PSO, que no caso desta proposta de estudo tem o espalhamento como característica, para gerar convergência e adaptação às modificações do sistema.

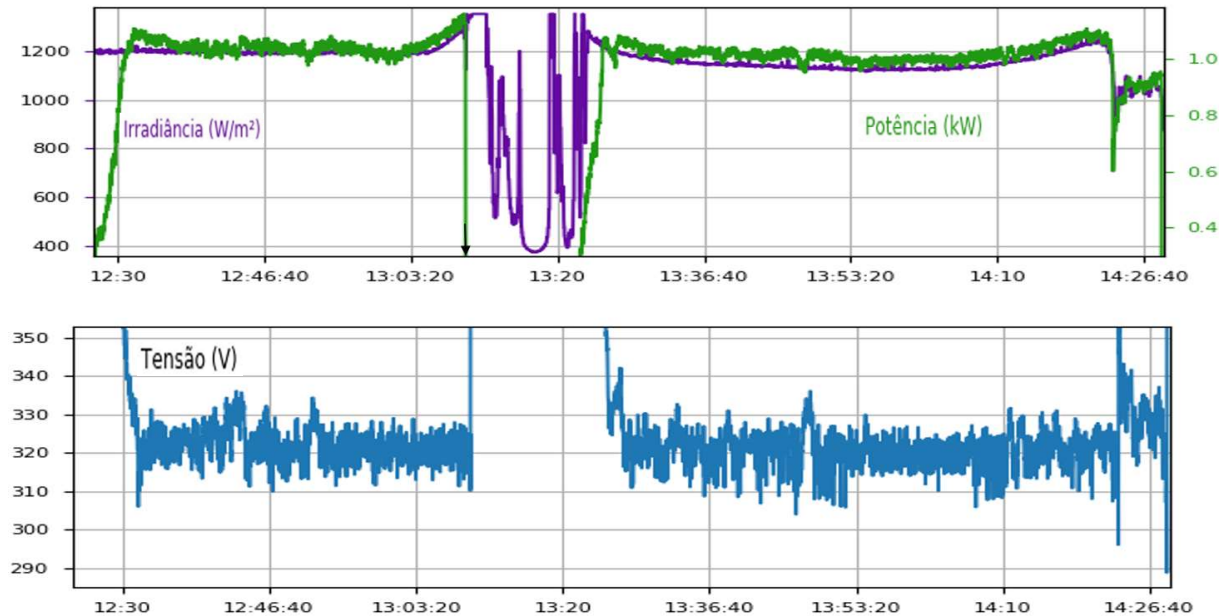


Figura 18: Gráficos de potência x Irradiância e tensão, entre 12:30 e 14:26 horas

Fonte: Elaborado pelo autor

Contudo, durante o período das 14:23 às 15:07, com variações de irradiância, verifica-se que há variações de mais de 60V no sinal de tensão durante a busca por convergência, isso se dá pois, conforme apresentado no subitem 3.2.1, mesmo que o valor de *setpoint* sendo empiricamente escolhido, quando a tensão assume valores menores que 290 V as partículas assumem valores em torno de 80% e 90% do valor do melhor global, isso pode significar diminuições bruscas de sinal de controle que, conseqüentemente, ocasiona diminuição brusca da corrente e potência demandadas do sistema, fazendo com que a tensão se eleve consideravelmente, entretanto, por conta disso, não há erro de subtensão e o controle pode voltar a atuar e convergir.

Pode-se verificar que durante o período citado houve uma grande diminuição na capacidade de convergência das partículas, isso se dá pelo fato de que a intensa variação de irradiância impede que haja a correta adequação dos valores do melhor local e global do

método, com isso, as partículas acabam por ter dificuldade em acompanhar as variações de tensão resultantes.

Nesse período, também, houve a maior quantidade de interrupções por subtensão, mostrando que a metodologia utilizada não possui resposta dinâmica o suficiente para acompanhar às bruscas variações de irradiância, conforme se observa pela figura 19.

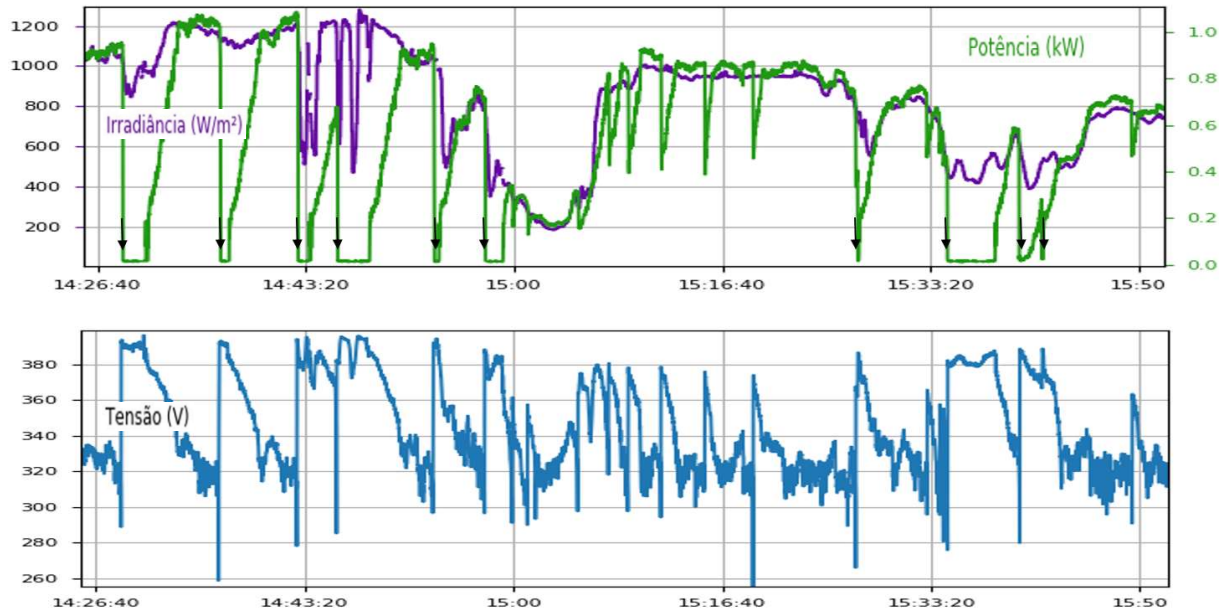


Figura 19: Gráficos de potência x Irradiância e tensão, entre 14:26 e 15:50 horas

Fonte: Elaborado pelo autor

Para tentar melhorar o comportamento mostrado acima foi realizado a implementação do método de realocação das posições das partículas, presente no pseudocódigo do Apêndice B nas linhas 38 a 42. Contudo, a metodologia de realocação das partículas para medições de tensão inferiores a 290V se apresenta como alternativa, pois causa afundamento de corrente e consequentemente de potência, em detrimento de evitar o erro de subtensão, por conta disso e da tensão de *setpoint* ser relativamente alta, pode se dizer que este é um método de controle conservador com perdas de potência relacionadas à própria metodologia.

Já quanto ao período final da aplicação do controle, após as 16:45 horas, verifica-se que conforme foi diminuindo o valor da irradiância, houve um aumento na variação do valor de tensão ao orbitar o valor desejado, gerando variações de até 40V para mais e 30V para menos.

Isso ocorre, pois, a diminuição da irradiância leva à diminuição da potência que pode ser entregue pelo GFV, e consequentemente, as alterações no sinal de controle provocam variações mais elevadas no sinal de tensão resultante, este aumento na sensibilidade da resposta

de tensão gera uma dificuldade de acomodação dessa tensão, conforme apresentado pela figura 20.

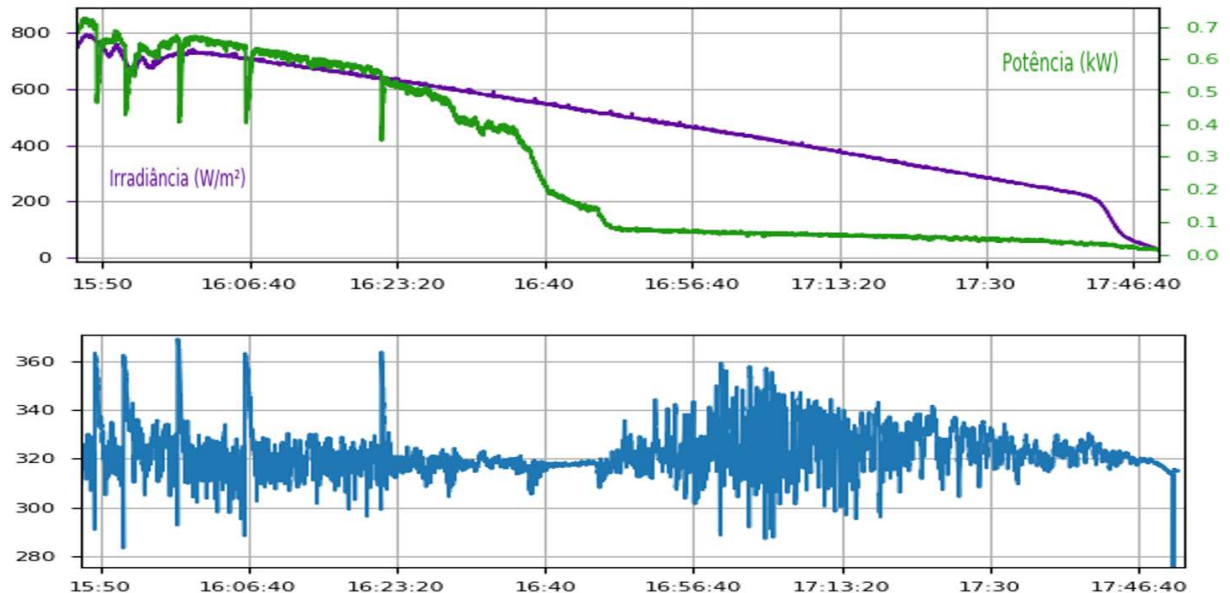


Figura 20: Gráficos de potência x Irradiância e tensão, entre 15:50 e 17:46 horas

Fonte: Elaborado pelo autor

Conforme Santos e Melo (2015) as causas de sombreamento e consequente afundamento de tensão, que são as variações de irradiância incidente sobre o módulo, são características advindas da própria resposta do GFV ao aparecimento de nuvens.

Verifica-se pelos resultados apresentados que este controlador PSO na metodologia proposta por este trabalho não apresenta resposta robusta ao sistema aplicado, devendo-se buscar uma forma de melhorar a resposta transitória, contudo, ainda cabe ressaltar que a operação da bancada de ensaios de SFB não ocorreu de maneira devida, sendo utilizados a motobomba de 2 cv mas fora da margem de operação adequada.

Neste teste foram verificados um total de 11 erros por subtensão, sendo indicados por setas pretas nas figuras 18 e 19, em que seus respectivos horários são relacionados na tabela 5.

Tabela 05: quantidade de interrupções do sistema por erro de subtensão no CF e horários em horas

1	2	3	4	5	6	7	8	9	10	11
13:09:30	14:28:40	14:36:28	14:42:42	14:45:54	14:53:36	14:57:41	15:27:30	15:34:47	15:40:25	15:42:13

Fonte: Elaborado pelo autor

Outro aspecto a ser considerado é que não há acomodação das partículas quando o controle converge para o *setpoint*, ou seja, as partículas continuam em constante estado de

busca, isso ocorre por conta das variações de irradiância e da constante necessidade de espalhamento para que haja convergência para os valores corretos de tensão. Por conta disso, pode-se dizer que esta é uma característica do próprio método, que conseqüentemente impede que as partículas se fixem em um sinal de controle e acomodação da tensão no valor do *setpoint*.

Os resultados apresentados levam em conta uma parametrização enfatizada na resposta mais rápida do sistema para variações de tensão, ou seja, escolheu-se por adotar como 3 o número de partículas; atraso de 600ms entre aplicação do sinal de controle e a leitura do sinal de tensão; o parâmetro “atualizar” igual a 2, ou seja, serão realizadas duas avaliações para cada uma das 3 partículas antes da atualização dos valores de referência, resultando em 6 iterações para tal atualização. Dessa forma, o sistema tem uma rápida resposta a variações, contudo verifica-se que a acomodação é prejudicada.

É sabido que cada problema de otimização responde de forma diferente para cada método utilizado para sua otimização, sendo buscado o melhor ou mais eficiente (SUN, LAI E WU, 2012), o próprio PSO possui diversas modificações e abordagens. Conseqüentemente, o algoritmo mostra resultados diversos conforme a modificação de seus parâmetros, sendo assim, sua correta parametrização é necessária para se obter os melhores resultados práticos.

Além disso, a aplicação de método de otimização como controle adaptativo gera uma gama de considerações adicionais na aplicação real, já que os métodos de otimização, no caso, o PSO, são desenvolvidos para aplicações puramente computacionais em que por muitas vezes são utilizadas funções de referência estáticas para o seu teste, conforme se verifica pelo trabalho de Serapião (2009).

Dentre as considerações a serem ponderadas estão: o tempo de aplicação do sinal de controle para se obter o sinal de resposta, no caso deste trabalho, o tempo de atraso entre aplicação do sinal e medição da tensão correspondente; a constante modificação das características do sistema, como a irradiância no caso do SFB, diferente de uma função estática utilizada em testes computacionais; dentre outras complicações dos sistemas reais.

Diante do exposto, verifica-se que testes adicionais são necessários para verificar a viabilidade de emprego dessa técnica da forma como foi proposta, pois o controlador PSO proporcional não se mostra capaz de responder de forma robusta, até por conta do SFB operar fora do ideal, sem altura manométrica e sem estudo do valor ideal da tensão para *setpoint* fixo.

5 CONCLUSÃO

A metodologia que foi aplicada é uma proposta inovadora do uso do PSO como forma de controle de um sistema real, que se dá, no caso, pelo uso de DPCM e bancada de ensaios de SFB presentes na UNIFAP, que por problemas técnicos não pode ser utilizada de maneira ideal. Por conta disso, foi cumprido o objetivo geral deste trabalho com a utilização do controlador PSO no SFB, através da bancada de ensaios de SFB da UNIFAP, com a ressalva de que a operação do sistema não foi a adequada.

Ademais, cumpriu-se os objetivos específicos de introduzir o PSO como método principal de controle, com a metodologia abordada no capítulo 3; parametrizar o algoritmo do PSO para obter o comportamento mais favorável ao sistema, no subitem 3.2.1; e analisar seu comportamento, no capítulo 4.

Com base no resultado do ciclo de medição apresentado no capítulo 4, é possível observar o funcionamento da metodologia, que para a aplicação em um SFB foi parametrizado para atuar de modo a evitar erros de subtensão do CF, com *setpoint* empírico de 320V e rotinas para evitar o erro de subtensão, trabalhando fora do ponto de máxima potência do sistema. Contudo, ainda não sendo robusto o suficiente para dar a melhor resposta para a característica de despacho intermitente de potência, mas durante o período de maior incidência da energia solar houve convergência.

Nesse sentido, sugere-se para trabalhos futuros a utilização de controlador PSO para plantas mais simples, que talvez apresente resultados mais favoráveis de resposta transitória para sistemas com variações mais suaves; a produção de um algoritmo; a utilização outras variações do PSO, pois o algoritmo utilizado foi uma variação do PSO clássico com poucos parâmetros adicionados, sendo que talvez uma metodologia mais sofisticada possa resolver os problemas encontrados no capítulo 4

REFERÊNCIAS

- AL-DUNAINAWI, Yousif; ABBOD, Maysam F.; JIZANY, Ali. A new MIMO ANFIS-PSO based NARMA-L2 controller for nonlinear dynamic systems. **Engineering Applications of Artificial Intelligence**, v. 62, p. 265-275, 2017.
- ALONSO ABELLA, M.; LORENZO, E.; CHENLO, F. PV water pumping systems based on standard frequency converters. **Progress in Photovoltaics: Research and Applications**, v. 11, n. 3, p. 179-191, 2003.
- BRITO, A. U. **Otimização do acoplamento de geradores fotovoltaicos a motores de corrente alternada através de conversores de frequência comerciais para acionar bombas centrífugas**. Tese (Doutorado em Energia) – Programa de Inter unidades de pós-graduação em Energia, Universidade de São Paulo. São Paulo, 2006.
- CAPELLI, Alexandre. **Inversores de Frequência**. Revista Mecatrônica Atual, São Paulo, nº 2, p. 7 – 15. Fev. 2002. Suplemento.
- CARNEIRO, Joaquim. **Módulos fotovoltaicos: características e associações**. Universidade do Minho. Azurém, 2010.
- COSTA, A. F.; GUIMARÃES JUNIOR, S. Dispositivo para Programação de Controladores e seu uso em Conversores de Frequência. Tese (Bacharelado em Engenharia Elétrica) – Universidade Federal do Amapá. Macapá, 2017.
- CRESESB. **Manual de Engenharia para Sistemas Fotovoltaicos**. Grupo de trabalho de Energia Solar – GTES. Edição revisada e atualizada, Rio de Janeiro, 2014.
- DE MANEIRA. **Como funciona o conversor de frequência**. Mundo da elétrica. 201-. Disponível em: <<https://www.mundodaeletrica.com.br/como-funciona-o-inversor-de-frequencia/>>. Acesso em: 27 Mar. 2019.
- EBERHART, Russell; KENNEDY, James. Particle swarm optimization. In: **Proceedings of the IEEE international conference on neural networks**. 1995. p. 1942-1948.
- EBERHART, Russell; SHI, Yuhui. Comparing inertia weights and constriction factors in particle swarm optimization. In: **Proceedings of the 2000 congress on evolutionary computation. CEC00 (Cat. No. 00TH8512)**. IEEE, 2000. p. 84-88.
- GARCIA, E. P.; ONAKA, J. M. D. **Uso De Técnicas de Otimização por Enxame de Partículas Aplicadas na Alocação e Dimensionamento Ótimo de Banco Capacitores em Sistemas de Distribuição de Energia Elétrica**. Tese (Bacharelado em Engenharia Elétrica) – Universidade Federal do Amapá. Macapá, 2016.
- LOPES, Felipe Rebelo. SINTONIA DE CONTROLADOR PID UTILIZANDO OS MÉTODOS R2W E PSO. **Engevista**, v. 19, n. 2, p. 482-497, 2017.
- MARANHÃO, G. N. A. **Controlador Fuzzy para uso em Conversores de Frequência aplicado a sistemas fotovoltaicos de Bombeamento**. Tese (Doutorado em Engenharia

Elétrica) – Programa de pós-graduação em Engenharia Elétrica, Universidade Federal do Pará. Belém, 2015.

MARANHAO, Geraldo NA et al. Experimental results of a fuzzy controlled variable-speed drive for photovoltaic pumping systems: A review. **IEEE Sensors Journal**, v. 16, n. 9, p. 2854-2864, 2016.

MARANHÃO, G. N. A.; MODESTO, C. S. Uso de PSO Modificado para Controle Dinâmico de Motor CC. 12th CLAGTEE, Mar del Plata, Argentina, 2017.

MIRJALILI, Seyedali; MIRJALILI, Seyed Mohammad; LEWIS, Andrew. Grey wolf optimizer. **Advances in engineering software**, v. 69, p. 46-61, 2014.

MPPSOLAR. **Ligação em série de mais painéis solares**. 2011. Disponível em: www.mpptsolar.com/pt/paineis-solares-em-paralelo.html. Acesso em: 27 Mar. 2019.

OGATA, Katsuhiko; YANG, Yanjuan. **Modern control engineering**. Prentice-Hall, 2002.

PEROZA, Josiane. **Caracterização elétrica de módulos fotovoltaicos de distintas tecnologias a partir de ensaios com simulador solar e iluminação natural**. Universidade Federal de Santa Catarina – Campus Araranguá. Araranguá, 2015.

ROCHA, Kamila P. et al. Controle de sistemas não lineares: um estudo de caso na graduação. **Leopoldina–MG**, 2012.

SAHW, Probeer; GAUR, Prerna. Photovoltaic Powered Centrifugal Water Pump. **International Journal of Electronic and Electrical Engineering**, v. 7, p. 247-254, 2014.

SANTOS, F. M; MELO, F. D. S. **Introdução aos conceitos e estudos de viabilidade técnico-econômica para implantação de uma central de geração de energia, eólico-solar, a ser introduzida na rede básica**. Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2015.

SERAPIÃO, Adriane Beatriz de Souza. Fundamentos de otimização por inteligência de enxames: uma visão geral. **Sba: Controle & Automação Sociedade Brasileira de Automatica**, v. 20, n. 3, p. 271-304, 2009.

SHI, Yuhui et al. Particle swarm optimization: developments, applications and resources. In: **Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No. 01TH8546)**. IEEE, 2001. p. 81-86.

SHI, Yuhui; EBERHART, Russell. A modified particle swarm optimizer. In: **1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (Cat. No. 98TH8360)**. IEEE, 1998. p. 69-73.

SUN, J.; LAI, C.; WU, X. **Particle swarm optimisation: classical and quantum perspectives**. Boca Raton, FL: CRC Press, 2012.

TEO, Kenneth Tze Kin et al. Particle swarm optimization based maximum power point tracking for partially shaded photovoltaic arrays. **IEEE Transactions on Energy Conversion**, v. 27, p. 1027-1035, 2013.

VALENTE, M. A. S. **Caracterização automática de um Pannel Fotovoltaico**. Dissertação (Mestrado em Engenharia Eletrotécnica e Computadores – Faculdade Ciências e Tecnologia da Universidade Nova de Lisboa. Lisboa, 2011.

WEG S/A. **CFW08 Inversor de Frequência. Manual do Usuário**. Disponível em: www.weg.net, 2003. Acesso em: 06 de junho de 2019.

WEG S/A. **CFW10 Inversor de Frequência. Manual do Usuário**. Disponível em: www.weg.net, 2011. Acesso em: 06 de junho de 2019.

WEG S/A. **Motores de indução alimentados por inversores de frequência PWM**. Guia Técnico. Disponível em: www.weg.net, 2010. Acesso em: 06 de junho de 2019.

APÊNDICE A – Algoritmo para programação do Microcontrolador Mestre

Algoritmo produzido pelo autor para o Arduino IDE (Integrated Development Environment), de acordo com o fluxograma de programação do núcleo Mestre do DPCM, mostrado na figura 10, no subitem 2.3.2:

```
#include "SPI.h" //SdCard
#include "SD.h" //SdCard
#include "Wire.h"//I2C
#include "U8glib.h"
//Initialize objects
Sd2Card SDcard; //SdCard - (SDA -> A4, SCL -> A5)
U8GLIB_SSD1306_128X64 u8g(U8G_I2C_OPT_NO_ACK); //DisplayOled
//AnalogValue[0->Tensão, 1-> Corrente, 2->Vazão, 3-> Irradiância, 4-> Temperatura]
//AnalogValue[0->voltage, 1-> current, 2->waterflow, 3->irradiance, 4-> temperature]
float AnalogValue[5] = {0, 0, 0, 0, 0};
byte chipSelect = 4; //Select the digital port to assign to sd module
byte DS1307_ADDRESS = 0x68; //address rtc
byte zero = 0x00;
bool sdState = false, serialState = false;
void setup() {
  Wire.begin(); //Begins the I2C communication
  Serial.begin(9600); //Begins the serial communication
  //Initiate sdcard
  //Tries to initiate the sdcard module
  //lambda expression that shows the states for sd inicialization
  auto msg = [] (String str) {
    u8g.firstPage();
    do {
      u8g.setFont(u8g_font_gdr12);
      u8g.setPrintPos(2, 32);
      u8g.print(str);
    }
    while (u8g.nextPage());
    delay(1000);
  };
  msg("Iniciando Sd...");
```

```

if (!SD.begin(chipSelect)) {
    msg("Falha ao Iniciar :(");
    sdState = false;
}
else {
    msg("Sd Iniciado :)");
    sdState = true;
    return;
};
}

void loop() {
    //Get physical value from transducers
    //MeansZeroFromAnalogPort(uint8_t analogPort, uint16_t iterations, float conversionConstant)
    /*AnalogValue[0] = MeansZeroFromAnalogPort(A1, 50, 500 / 1024); //Voltage
    AnalogValue[1] = MeansZeroFromAnalogPort(A0, 50, 10 / 1024); //Current
    AnalogValue[2] = MeansZeroFromAnalogPort(A3, 50, (5 * 14) / 1024) - 8; //Water flow
    AnalogValue[3] = IrradianceFromWire(); //Irradiance
    AnalogValue[4] = MeansZeroFromAnalogPort(A2, 50, 4.8 / (1024 * 0.1)); //Temperature*/
    /* For test the behaviours/debug*/
    AnalogValue[0] = float(randômico entre(0, 500)) + float(randômico entre(0, 10)) / 10;
    AnalogValue[1] = float(randômico entre(0, 50)) + float(randômico entre(0, 10)) / 10;
    AnalogValue[2] = float(randômico entre(0, 100)) + float(randômico entre(0, 10)) / 10;
    AnalogValue[3] = float(randômico entre(0, 1000)) + float(randômico entre(0, 10)) / 10;
    AnalogValue[4] = float(randômico entre(0, 45)) + float(randômico entre(0, 10)) / 10;
    //Send to IHM
    //SendSerial(separator, voltage, current, waterflow, irradiance, temperature)
    SendSerial('_', AnalogValue[0], AnalogValue[1], AnalogValue[2], AnalogValue[3],
AnalogValue[4]);
    //Get from RTC module year, month, day of the month, day of the week, hour, minute and second
    // _seg->second, _min->minute, _hor-hour, dsem->dayoftheweek, dmes->dayofthemouth
    // _mes->mouth, _ano->year
    uint8_t _seg, _min, _hor, dsem, dmes, _mes, _ano;
    Wire.beginTransmission(DS1307_ADDRESS);
    Wire.write(zero);
    Wire.endTransmission();
    Wire.requestFrom(DS1307_ADDRESS, 7);//
    _seg = ToDecimal(Wire.read());

```

```

_min = ToDecimal(Wire.read());
_hor = ToDecimal(Wire.read() & 0b111111);
dmes = ToDecimal(Wire.read());
dmes = ToDecimal(Wire.read());
_mes = ToDecimal(Wire.read());
_ano = ToDecimal(Wire.read());
//Write, the data received, to sd card
File dataFile = SD.open("DADOS.TXT", FILE_WRITE);
if (dataFile)
{
  if (dataFile.size() == 0) {
    //Header
    //If the file is being created for the first time then this apply it, else
    //only the data will be write. This snippet ensures that the output is according
    // to csv standard.
    dataFile.println("Data\tHora\tTensão\tCorrente\tVazão\tIrradiância\tTemperatura");
  }
  dataFile.print(String(dmes) + "/" + String(_mes) + "/" + String(_ano) + "\t"); //write date
  dataFile.print(String(_hor) + ":" + String(_min) + ":" + String(_seg) + "\t"); //write hour
  //write physical values from
  dataFile.print(String(AnalogValue[0]) + "\t"); //voltage
  dataFile.print(String(AnalogValue[1]) + "\t"); //current
  dataFile.print(String(AnalogValue[2]) + "\t"); //water flow
  dataFile.print(String(AnalogValue[3]) + "\t"); //irradiance
  dataFile.println(String(AnalogValue[4])); //temperature
  dataFile.close();
}
else { }
//Display in oled display
// picture loop
u8g.firstPage();
do {
  draw(AnalogValue[0], AnalogValue[1], AnalogValue[2], AnalogValue[3], AnalogValue[4]);
} while ( u8g.nextPage() );
serialState = false;
delay(800);
}

```



```

byte ToDecimal(byte val) {
    //Convert BCD to decimal
    return ( (val / 16 * 10) + (val % 16) );
}

void draw(float voltage, float current, float waterFlow, float irradiance, float temperature) {
    //This function draw on oled display
    // Voltage
    u8g.setPrintPos(60, 12);
    u8g.print("Ten:");
    u8g.setPrintPos(85, 12);
    u8g.print(floatToString5(voltage));
    // Current
    u8g.setPrintPos(60, 24);
    u8g.print("Cor:");
    u8g.setPrintPos(85, 24);
    u8g.print(floatToString5(current));
    // Water flow
    u8g.setPrintPos(60, 36);
    u8g.print("Vaz:");
    u8g.setPrintPos(85, 36);
    u8g.print(floatToString5(waterFlow));
    // Irradiance
    u8g.setFont(u8g_font_fixed_v0);
    u8g.setPrintPos(60, 48);
    u8g.print("Irr:");
    u8g.setPrintPos(85, 48);
    u8g.print(floatToString5(irradiance));
    // Temperature
    u8g.setPrintPos(60, 60);
    u8g.print("Tem:");
    u8g.setPrintPos(85, 60);
    u8g.print(floatToString5(temperature));
    u8g.drawFrame(3, 14, 47, 29);
    String t1 = "Sd:";
    String t2 = "Se:";
    String s1 = "ok";
    String s2 = "fail";
}

```

```

if (sdState == true) {
    u8g.setPrintPos(6, 24);
    u8g.print(t1 + s1);
} else if (sdState == false) {
    u8g.setPrintPos(6, 24);
    u8g.print(t1 + s2);
}
if (serialState == true) {
    u8g.setPrintPos(6, 39);
    u8g.print(t2 + s1);
} else if (sdState == false) {
    u8g.setPrintPos(6, 39);
    u8g.print(t2 + s2);
}
}

// Convert a floating number to string with 6 caracteres, e.g,
// float x = 1.3012315123 -> floatToString5(x) = "1.3012"
// float y = 1.2 -> floatToString5(y) = "1.2000"
String floatToString5(float input)
{
    String output, aux = String(input, DEC);
    for (int i = 0; i <= 5; i++) {
        output.concat(aux[i]);
        if (aux[i] = 0) output.concat('0');
    }
    return output;
}

/*Function responsible for send to serial interface the data that will be shown in the IHM for desktop*/
void SendSerial(char separator, float voltage, float current, float waterFlow, float irradiance, float
temperature)
{
    String str = "";
    str += floatToString5(voltage);
    str += separator;
    str += floatToString5(current);
    str += separator;
    str += floatToString5(waterFlow);
}

```

```

str += separator;
str += floatToString5(irradiance);
str += separator;
str += floatToString5(temperature);
Serial.println(str);
serialState = true;
}
/* Functions responsible for handle the values incoming from transducers */
//Return an analog value from an 'analogPort' performing a Mean with N(iterations) and using
//the 'conversionConstant' for converting the output according to physical value measured
float MeansZeroFromAnalogPort(uint8_t analogPort, uint16_t iterations, float conversionConstant) {
    float sum = 0;
    for (int i = 0; i < iterations; i++) {
        sum = sum + analogRead(analogPort);
    }
    return (sum / iterations) * conversionConstant;
}
float IrradianceFromWire() {
    Wire.requestFrom(8, 4);
    char c;
    float sum = 0 , IrradianceValue = 0;
    int i = 1000; //inicia com milhar
    while (Wire.available()) {
        c = Wire.read(); //ler o barramento I2C
        IrradianceValue = c - '0'; //converte caractere em inteiro
        sum = sum + IrradianceValue * i; //a variavel i serve para organizar as casas.
        i = i / 10; //reduz para centena, dezena e unidade.
    }
    return (110.5 - (sum * (5 / 1024) * 28.80541) ) * (1000 / 76.53);
}

```

APÊNDICE B – Algoritmo para programação do Microcontrolador Escravo

Pseudocódigo e algoritmo produzido para o Arduino IDE (Integrated Development Environment), de acordo com o fluxograma de programação do núcleo Escravo do DPCM, mostrado na figura 09, no subitem 2.3.2:

```
#include "ControleFuzzy.h"
#include <Wire.h>
//FuzzyContrl(uint_8 AnalogPortIrradiance);
//Constructor that initiate the analog port for irradiance input
FuzzyControl fuzzy(A2);
int atraso = 600;
int setpoint = 320; //320 volts -> valor desejado de tensão
float C = 2, S = 2.7; // Coeficiente cognitivo e coeficiente social
float fi = C + S; //fi deve ser > 4
float K = 2 / abs(2 - fi - pow((pow(fi, 2) - 4 * fi), 0.5)); // Fator de constição
float w = 0.9; // Cte de inércia
int N = 3; //nº de Partículas
float erro = setpoint; // inicialização do valor de erro
float x[10], v[10], xlocal[10], flocal[10], fglobal = 0, xglobal = 0, vv, xx, VT;
byte aux_atualizar = 0, atualizar = 2; //auxiliares de espalhamento
float local, global;
byte sinal_controle = 3; //Pino de aplicação do sinal de controle
byte volt_in = A0; //Leitura do sinal de tensão -> porta analogica A0
void setup() {
    //As partículas reinicializam se o sistema reiniciar
    //Inicialização das partículas, velocidades e valores ótimos
    for (int i = 0; i <= N; i++)
    {
        x[i] = 90 + randômico entre(50, 100) / 10.0;
        v[i] = randômico entre(10, 100) / 10.0;
        xlocal[i] = x[i]; // iniciação do melhor individual da partícula
        flocal[i] = 0;
    }
    //=====Inicia o barramento I2C=====
    Serial.begin(9600);
    Wire.begin(8);
    Wire.onRequest(requestEvent);
```

```

}
void loop() {
  //Measures the irradiance at AnalogPort set on constructor
  //trecho de código destinado às rotinas para leitura de irradiância
  fuzzy.MeasureIrradiance(5, 0.85, 1000.0 / (0.751 * 1024.0) );
  // Buscar limitação de x nos valores de PWM entre 0 e 255
  // buscar o valor da tensão de alimentação da bomba
  for (int i = 0; i < N; i++)
  {
    local = C * (randômico entre(1, 100) / 100.0) * (xlocal[i] - x[i]);
    global = S * (randômico entre(1, 100) / 100.0) * (xglobal - x[i]);
    vv = K * (w * v[i] + local + global);
    xx = x[i] + vv;
    x[i] = xx;
    v[i] = vv;
    //Critério de restrição contra divergências
    if (x[i] > 255 || x[i] < 0 || v[i] > 255)
    {
      x[i] = randômico entre((xglobal - 5) * 10 , (xglobal + 5) * 10) / 10.0; //x assume um valor aleatório
em torno de xglobal
      vv = randômico entre(v[i] * 7, v[i] * 13) / 10.0;
      v[i] = vv; //v assume um randômico entre 70% e 130% V[i]
    }
    //Aplicar a avaliação, fitness
    analogWrite(sinal_controle, x[i]); //aplicar valor da partícula
    delay(atraso);
    VT = 0;
    for (int j = 0; j < 50; j++)
      VT += (float)analogRead(volt_in) * 500.0 / 1023.0 * 0.86; //ler valor de tensão
    VT = VT / 50; //VT assume a média de 50 valores instantâneos lidos
    if (VT < 290) { //forma de evitar afundamento de tensão
      for (int j = 0; j < N; j++)
        x[j] = randômico entre((xglobal*8) , (xglobal*9)) / 10.0;
      goto avaliar;
    }
    // Atualizar o valor social/global da partícula
    erro = abs(VT - setpoint);

```

```

if (erro <= abs(fglobal - setpoint)) {
    xglobal = x[i];
    fglobal = VT;
}
// Atualizar o valor cognitivo/local da partícula
if (erro <= abs(flocal[i] - setpoint)) {
    xlocal[i] = x[i];
    flocal[i] = VT;
}
}
//estratégia para espalhamento das partículas e atualização de xlocal e x global
if (aux_atualizar >= atualizar) {
    if (VT - setpoint < 20) { //limitação do espalhamento
        for (int j = 0; j < N; j++)
            x[j] = randômico entre((xglobal - 2) * 10 , (xglobal+1) * 10) / 10.0;
    } else {
        for (int j = 0; j < N; j++)
            x[j] = randômico entre((xglobal - 2) * 10 , (xglobal + 5) * 10) / 10.0;
    }
    avaliar:
    fglobal = 0;
    for (int c = 0; c < N; c++) {
        analogWrite(sinal_controle, x[c]); //aplicar valor da partícula
        delay(atraso);
        VT = 0;
        for (int j = 0; j < 50; j++)
            VT += (float)analogRead(volt_in) * 500.0 / 1023.0 * 0.86; //ler valor de tensão
        VT = VT / 50;
        if (VT < 290) { //forma de evitar afundamento de tensão
            for (int j = 0; j < N; j++)
                x[j] = randômico entre((xglobal*8) , (xglobal*9)) / 10.0;
            goto avaliar;
        }
        xlocal[c] = x[c]; //atualização do melhor local
        flocal[c] = VT;
        if (abs(flocal[c] - setpoint) <= abs(fglobal - setpoint)) {
            xglobal = x[c]; //atualização do melhor global

```

```

    fglobal = flocal[c];
}
    local = C * (randômico entre(1, 100) / 100.0) * (xlocal[c] - x[c]);
    global = S * (randômico entre(1, 100) / 100.0) * (xglobal - x[c]);
    vv = K * (w * v[c] + local + global);
    v[c] = vv;
}
    aux_atualizar = 0;
}
    aux_atualizar++;
}
//=====Envio da Informação pelo Barramento I2C=====
void requestEvent() {
    //convert para string
    char c[5];
    String str;
    str = fuzzy.GetIrradiance();
    //colocar 0 em valores para manter 4 digitos para valores menores que 1000
    String c1 = "0", c2 = "00", c3 = "000";
    if (fuzzy.GetIrradiance() <= 10)
    {
        //acrescenta o "0"
        c3.concat(str);
        //converte para char
        c3.toCharArray(c, 5);
    }
    else
    {
        if (fuzzy.GetIrradiance() < 100)
        {
            //acrescenta o "0"
            c2.concat(str);
            //converte para char
            c2.toCharArray(c, 5);
        }
        else
        {

```

```

if (fuzzy.GetIrradiance() < 1000)
{
    //acrescenta o "0"
    c1.concat(str);
    //converte para char
    c1.toCharArray(c, 5);
}
else
{
    //converte para char
    str.toCharArray(c, 5);
}
}
}
//enviar a informação
Wire.write(c);
/*Serial.print("\nI2c:\t");
Serial.print(c);*/
delay(500);
}
String leStringSerial() {
String conteudo = "";
char caractere;
// Enquanto receber algo pela serial
while (Serial.available() > 0) {
// Lê byte da serial
caractere = Serial.read();
// Ignora caractere de quebra de linha
if (caractere != '\n') {
// Concatena valores
conteudo.concat(caractere);
}
// Aguarda buffer serial ler próximo caractere
delay(10);
}
return conteudo;
}
}

```

```

1  atraso ← 600
2  setpoint ← 320 Setpoint: valor desejado de tensão
3  C ← 2 C e S: Coeficientes cognitivo e social, respectivamente
4  S ← 2.7
5  fi ← C + S Fator de restrição e φ, descritos na equação 2.4
6  K ← 2 / |2 - fi - (fi^2 - 4 * fi)^0.5|
7  w ← 0.9 Constante de inércia
8  N ← 3 n° de partículas
9  aux_atualizar ← 0
10 atualizar ← 2 auxiliares de espalhamento e atualização
11 sinal_controle ← 3 Pino de aplicação do sinal de controle
12 volt_in ← A0 Pino que recebe o sensor de tensão
13 rotina inicial do microcontrolador(setup)
14   para i ← 0 até N faça Inicialização das partículas
15     x[i] ← 90 + (randômico entre 1 e 10)
16     v[i] ← (randômico entre 1 e 10)
17     xlocal[i] ← x[i]
18     flocal[i] ← 0
19 fim da rotina inicial
20
21 início do código em laço
22 para i ← 0 até N faça
23   local ← C*(randômico entre 0 e 1)*(xlocal[i] - x[i])
24   global ← S*(randômico entre 0 e 1)*(xglobal - x[i])
25   v[i+1] ← K * (w * v[i] + local + global)
26   x[i+1] ← x[i] + v[i+1]
27   limitação de x nos valores de PWM entre 0 e 255
28   se x[i] > 255 ou x[i] < 0 ou v[i] > 255 faça
29     x[i] ← randômico entre xglobal-5 e xglobal+5
30     v[i] ← randômico entre v[i] * 0.7 e v[i] * 1.3
31   fim se
32   aplicar na porta sinal_controle o sinal PWM x[i]
33     esperar atraso
34   VT ← 0
35   para j ← 0 até 50 faça
36     VT recebe e incrementa o valor lido da porta volt_in
37   VT ← VT / 50 recebe a média de 50 valores instantâneos
38   se VT < 290 faça Contra o afundamento de tensão
39     para j ← 0 até N faça
40       x[j] ← randômico entre xglobal*0.8 e xglobal*0.9
41
42     vá para a posição avaliar
43   fim se
44   erro ← |VT - setpoint|
45   se erro <= |fglobal - setpoint| faça
46     xglobal ← x[i]
47     fglobal ← VT
48   fim se
49
50   se erro <= |flocal[i] - setpoint| faça
51     xlocal[i] ← x[i]
52     flocal[i] ← VT
53   fim se
54
55
56 se aux_atualizar >= atualizar faça

```

```

57
58   se VT - setpoint < 20 faça
59     para j ← 0 até N faça
60       x[j] ← randômico entre xglobal-2 e xglobal+1
61 Senão faça
62   para j ← 0 até N faça
63     x[j] ← randômico entre xglobal-2 e xglobal+5
64 fim se
65
66 posição avaliar:
67
68     fglobal ← 0
69 para c ← 0 até N faça
70   aplicar na porta sinal_controle o sinal PWM x[c]
71
72     esperar atraso
73
74   VT ← 0
75 para j ← 0 até 50 faça
76   VT recebe e incrementa o valor lido da porta volt_in
77
78     VT ← VT
79
80 se VT < 290 faça
81   para j ← 0 até N faça
82     x[j] ← randômico entre xglobal*0.8 e xglobal*0.9
83
84   vá para a posição avaliar
85 fim se
86
87   xlocal[c] ← x[c]
88   flocal[c] ← VT
89   se |flocal[c]-setpoint| <= |fglobal-setpoint| faça
90     xglobal ← x[c]
91     fglobal ← flocal[c]
92   fim se
93
94   local ← C*(randômico entre 0 e 1)*(xlocal[c] - x[c])
95   global ← S*(randômico entre 0 e 1)*(xglobal - x[c])
96   v[c+1] ← K * (w * v[c] + local + global)
97   x[c+1] ← x[c] + v[c+1]
98   aux_atualizar = 0;
99 fim se
100 aux_atualizar se incrementa em 1
101 fim do laço

```

Pseudocódigo da programação do controlador PSO aplicado ao microcontrolador

Fonte: Elaborado pelo autor

APÊNDICE C – Simulação de controlador PSO proporcional em motor c.c.

Visando subsidiar a proposta do uso do PSO em problemas reais, este Apêndice apresenta a simulação do uso do PSO como método de controle adaptativo de um motor c.c., utilizando-se modificações dos parâmetros abordados no subitem 2.1 deste trabalho.

Esta abordagem é mostrada para compreensão do comportamento do PSO aplicado em ambiente simulado, na qual observou-se parâmetros como convergência, espalhamento das partículas, modificação do ponto de busca, dentre outras formas de entendimento do método.

A simulação realizada apresenta uma planta que responde com características de sistema de segunda ordem, o qual representa o comportamento de um motor c.c., sendo assim, o PSO é utilizado neste sistema na forma de um controlador proporcional, em que as partículas representam o ganho desse controlador (k_p) e o programa age de forma a modificar esse ganho para que mantenha a velocidade do motor constante em relação a modificações na sua alimentação ou perturbações externas.

Sendo assim, com o intuito geral de analisar o comportamento do PSO e verificar a convergência do método quando aplicado ao sistema de segunda ordem, utilizou-se as partículas como possíveis soluções, que se movem por um espaço vetorial unidimensional, este espaço é composto pelos possíveis valores de k_p , no caso, valores de posição da partícula.

Sabe-se que os sistemas de segunda ordem servem como modelo para que seja descrito sistemas reais, tanto para a sua aproximação, quanto para sua completa descrição, no caso de sistemas de ordens superiores (OGATA, 2002). Para obtenção de resultados, foi simulado no software MATLAB a resposta ao degrau unitário da planta de um motor c.c. tendo em sua malha de controle um controlador proporcional que tem seu ganho modificado pelo PSO.

De acordo com o apresentado por Ogata (2002), o motor c.c. tem um comportamento já bem definido e estudado, de modo que é sabido o modelo matemático que melhor o representa, o de um sistema de segunda ordem. Ora, tanto para um motor controlado por armadura quanto controlado por campo, as equações base que os descrevem são similares, mudando, por conseguinte, as variáveis de cada tipo de controle.

O principal modelo da função de transferência de um sistema de segunda ordem é aquele que apresenta pelo menos os parâmetros básicos para a descrição do sistema, são eles a frequência natural não amortecida (ω_n) e a constante de amortecimento (δ).

$$\frac{\omega_n^2}{s^2 + 2\delta\omega_n s + \omega_n^2}$$

(Apêndice C.1)

Este modelo simplificado de equação de segunda ordem, por satisfazer as necessidades para simulação, será adotado para representar o sistema a ser controlado, pois traz comodidade de parametrização para a programação.

Sendo assim, utiliza-se o diagrama de blocos simplificado do motor CC, que para fins de simplificação, conta com as principais constantes que caracterizam o comportamento da planta e nesse diagrama teórico será incluso o controlador proporcional (OGATA, 2002). Sendo apresentadas abaixo a equação dos parâmetros para o sistema, e o diagrama de blocos na figura a seguir:

$$\omega_n = \sqrt{\frac{K}{J}} \text{ e } \delta = \frac{B}{2\sqrt{JK}} \quad (\text{Apêndice C.2})$$

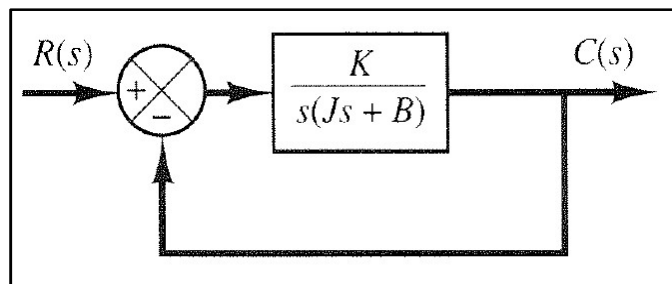


Figura: Diagrama de bloco simplificado

Fonte: OGATA, 2002, p. 120

Considerando uma resposta ao degrau unitário, a influência de um ganho K_p sempre modificará a resposta em regime permanente do sistema e pode alterar a oscilação e amortecimento, e para melhor apresentar o desempenho do algoritmo foi adotado o uso de sistema de segunda ordem subamortecido, já que essa característica é a que mais se aproxima à dinâmica de um motor c.c.

Será apresentada a resposta para três características distintas do sistema, de forma a mostrar o comportamento do algoritmo na variação da resposta, na forma mostrada na tabela abaixo:

Configurações	ω_n	δ	População	w
1	3	0,5	4	0.9
2	5	0,8	5	0.8
3	2	0,3	6	0.8

Tabela: Modificações nas características do sistema de segunda ordem

Fonte: Elaborado pelo autor

Os gráficos gerados foram obtidos pelo MATLAB através da plotagem da resposta do sistema de segunda ordem sem controlador, gerado uma curva suave de resposta, e do sistema com controlador, gerado uma resposta com dinâmica que se verifica a ação das partículas, operando ponto a ponto na ação de controle. As figuras 1, 2 e 3 mostram a resposta para as configurações do sistema, mostradas na tabela 3:

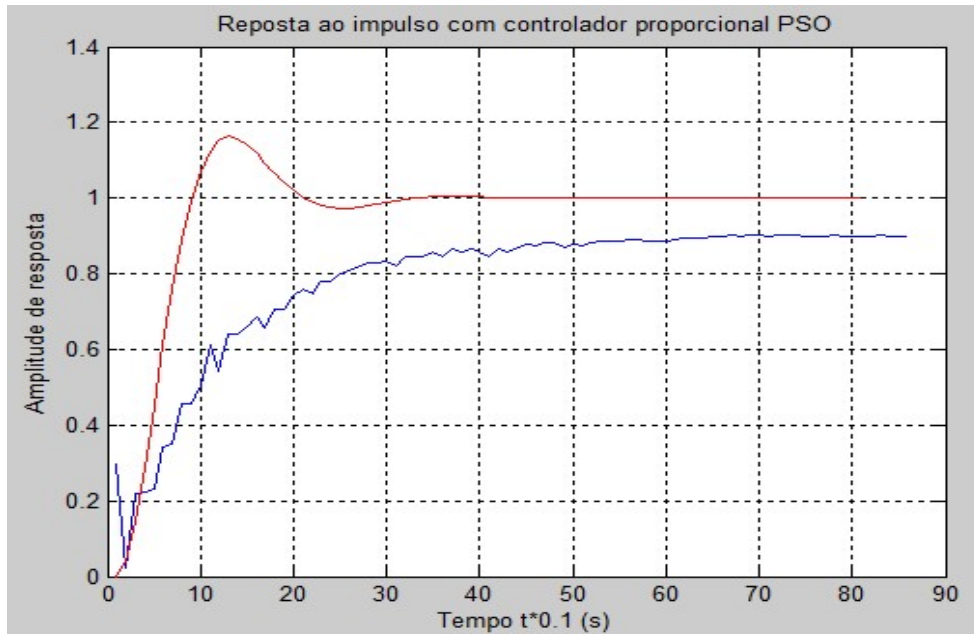


Figura: Comparação da configuração 1

Fonte: MARANHÃO; MODESTO (2017)

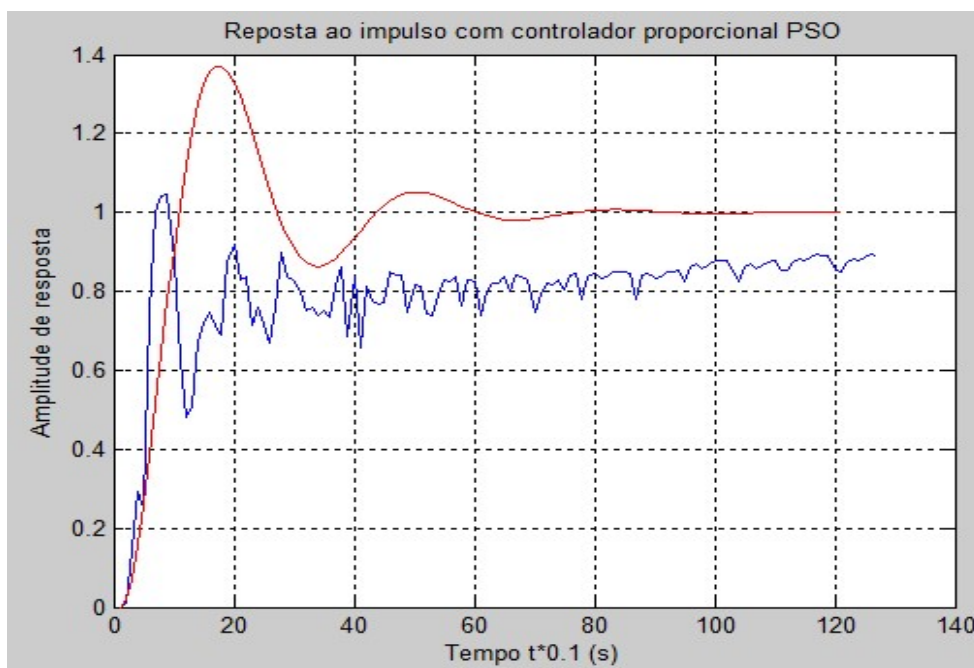


Figura: Comparação da configuração 2

Fonte: MARANHÃO; MODESTO (2017)

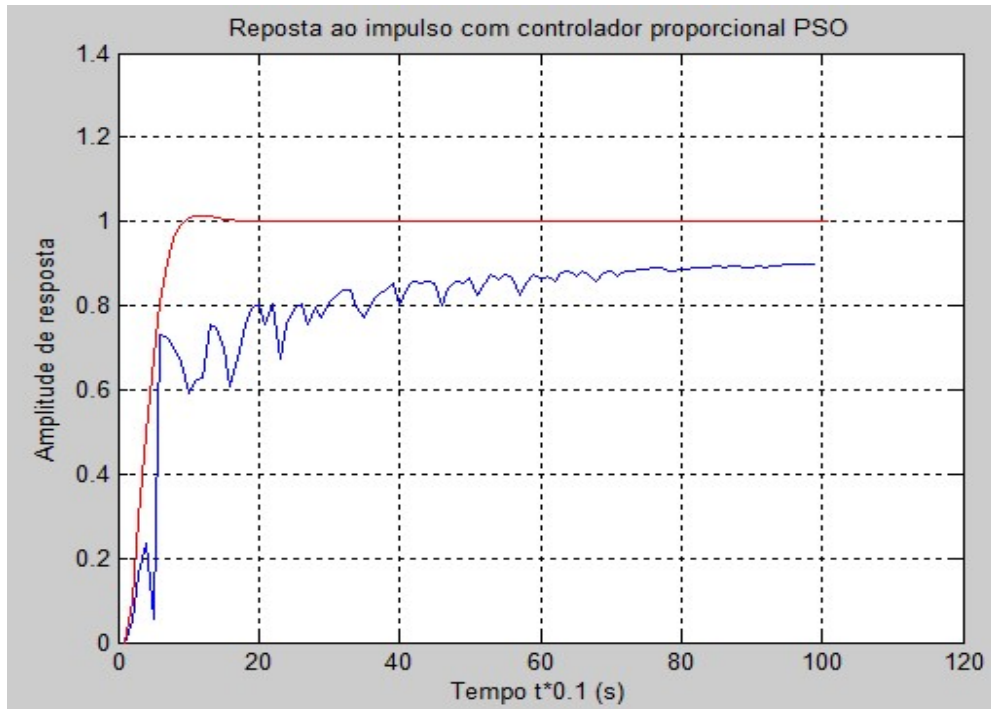


Figura: Comparação da configuração 3

Fonte: MARANHÃO; MODESTO (2017)

De forma a melhorar a apresentação dos resultados foi separado a sua abordagem na exposição do obtido em regime permanente, no transitório e as características de parametrização.

A resposta do sistema apresentado corresponde à velocidade de um motor c.c. e o valor de resposta um representa a resposta em regime permanente de um sistema sem controlador, então o algoritmo buscou de modo efetivo valores de resposta conforme o esperando, sendo que as partículas poderiam assumir valores de ganho proporcional K_p , que se insere na equação de controle na forma apresentada abaixo:

$$\frac{K_p}{1+K_p}$$

(Apêndice C.2)

Tendo em vista que o valor acima é a ação de um controlador proporcional sobre a saída em regime permanente, este valor seria próximo de 1 para valores muito altos de K_p . Dessa forma, para evitar a elevação do valor das partículas é determinado que as partículas respondam para alcançar o valor se 0,9 de amplitude, dessa forma resultando em um erro de regime permanente de 10%.

O erro de regime permanente se dá, portanto, pela necessidade de limitação dos valores das partículas (condição de restrição), sendo o valor do controlador a ser alcançado de $K_p = 9$, obtendo a resposta mencionada.

A resposta em regime transitório se mostrou mais amortecida que a resposta natural do sistema sem controlador, contudo se observado o comportamento em regime transitório de um sistema com controlador proporcional verifica-se que há sobressinal associado no caso em que o fator de amortecimento é inferior a 1 (um) (sistema subamortecido), e considerando que o programa ameniza esse sobressinal na proporção em que o valor da resposta desejada é mais próximo de 1, então obteve-se uma vantagem com o método.

Deve-se observar principalmente para o aumento do valor de K_p no período inicial de regime transitório, já que a constante pode provocar no aumento da oscilação e gerando sobressinal elevado, o que nos casos apresentados não foi observado, mostrando a capacidade de adaptação do método.

Quanto aos demais parâmetros de desempenho, como tempo de acomodamento por exemplo, não é possível de se ter um cálculo teórico já que o comportamento das partículas varia a cada nova iteração.

Observa-se que a convergência é bem mais árdua dependendo das características da dinâmica do sistema. Observa-se que conforme simulações do sistema proposto realizadas no software MATLAB existe uma parametrização do MPSO que possibilita sua utilização para o controle adaptativo desse sistema de modo que se verifica convergência à resposta ideal em regime permanente e que através da modificação de características do algoritmo pode-se obter várias propriedades de resposta do regime transitório.

Algoritmo produzido para a linguagem do software MATLAB:

```
%% Inicialização do Programa
clc;      %limpa area de trabalho
close all;
clear all; %Limpa dados do workspace
C1 = 2;   % Coeficiente cognitivo
C2 = 3;   % coeficiente social
fi = C1+C2;
K = 2/abs(2-fi-(fi^2-4*fi)^0.5);% Fator de constição
w = 0.9; %cte de inercia
Nparticulas = 5;
```

```

fbest = 100;    %valor da partícula do melhor global
k = 0;         %Contador de avaliações(fitness) do sistema
erro = 100;
F=zeros(1,Nparticulas);
%% Inicialização das Partículas, suas velocidades e GBest
for i = 1:Nparticulas %repetição de inicialização das partículas
    x(i) = rand;    %inicialização de velocidade e posição da partícula
    v(i) = 2*rand;
    p(i) = x(i);%iniciaçozação do melhor individual da partícula
    k = k + 1;
    [f(i), re(k)] = fitness_step(x(i),k);
    resp(k) = f(i);
    floca(i) = f(i); %valor da avaliação do melhor global
    if f(i) <= fbest
        Gbest = x(i);
        fbest = f(i);
    end
end

while erro > 0.0001
    %% Rotina do programa
    for i = 1:Nparticulas %Atualização das partículas
        vv = K*(w*v(i) + C1*rand*(p(i)-x(i)) + C2*rand*(Gbest-x(i)));
        xx = x(i) + vv;
        x(i) = xx;
        v(i) = vv;
        if x(i) > mean(x)*(1+(std(x))^0.1) %Critério de restrição contra divergência
            x(i) = mean(x);
            v(i) = K*(w*v(i) + C1*rand*(p(i)-x(i)) + (C2*rand*(Gbest-x(i))));
            xx = x(i) + vv;
            v(i) = vv;
        end
        k = k + 1;
        [f(i), re(k)] = fitness_step(x(i), k);

```



```

resp(k) = f(i);
if f(i) <= fbest %Atualizar o valor social da partícula
    Gbest = x(i);
    fbest = f(i);
end
if f(i) <= flocal(i) %Atualizar o valor cognitivo da partícula
    p(i) = x(i);
    flocal(i) = f(i);
end
end
% mod 06/06 era erro = mean(x), ou seja, retornava a média de x
erro = std(x); % erro é baseado no desvio padrão da resposta !!!
if k > 90
    erro = 0;
end
%% Estratégia para evitar defasagem dos valores de fbest e flocal
k = k+1;
aux = fbest;
[fbest, re(k)] = fitness_step(Gbest, k);
resp(k) = fbest;
% se o valor da avaliação de gbest mudar então muda-se os melhores locais
if aux ~= fbest %(aux > fbest+0.2) || (aux < fbest-0.2)
    for i = 1:Npartículas
        k = k+1;
        [flocal(i), re(k)] = fitness_step(p(i), k);
        resp(k) = flocal(i);
    end
end
end

F=[F;x];
end
plot(resp),grid,title('Erro à resposta do Sistema com controlador proporcional PSO');
xlabel('Tempo (T - período)*segundos'),ylabel('Amplitude')
figure

```

```

plot(re),grid,title('Reposta ao impulso com controlador proporcional PSO');
xlabel('Tempo t*0.1 (s)'),ylabel('Amplitude de resposta')
hold on
wn = 3; %Frequencia natural em rad/s
s = 0.5; %Fator de amortecimento
r=tf(wn^2,[1 2*s*wn wn^2]);
t = 0 : 0.1 : 8;
plot(step(r,t),'r')
figure
plot(F),grid,title('Valores assumidos de Kp(particulas)por iterações');

```

```

function [y, z] = fitness_step(x, c)
val = 0.9;
wn = 3; %Frequencia natural em rad/s
s = 0.5; %Fator de amortecimento
k = x; %Ganho Kp, recebe as particulas a cada avaliação
T = 0.1; %Período de resposta estipulado para cada partícula
p = c;
t = 0:T:c*T; %Tempo o qual será aplicado para avaliação das partículas
if c > 100
    p = (mod(c,100));
    t = 10:T:(10+p*T);
end
r=tf(wn^2,[1 2*s*wn wn^2]);
res = k*r/(1+k*r);
A = step(res, t); %Erro da resposta em relação á referencia 1
z = A(p);
y = abs(val-z);
end

```