



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JOSÉ TIAGO SILVÉRIO

**SIMULADOR DE CONTROLE AUTOMATIZADO DE GERAÇÃO DE
VAPOR.**

Assis

2012

JOSÉ TIAGO SILVÉRIO

**SIMULADOR DE CONTROLE AUTOMATIZADO DE GERAÇÃO DE
VAPOR.**

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis (IMESA) e Fundação Educacional do Município de Assis (FEMA), como requisito do Curso de Graduação de Ciência da Computação.

Orientador: Profº. Guilherme Cleva Farto.

Área de Concentração: Informática

Assis

2012

FICHA CATALOGRÁFICA

SILVÉRIO, José Tiago

Simulador de controle automatizado de geração de vapor /

José Tiago Silvério. Fundação Educacional do Município de Assis – FEMA – Assis
2012.

41 p.

Orientador: Prof^o. Esp. Guilherme de Cleva Farto.

Trabalho de Conclusão de Curso

Instituto Municipal de Ensino Superior de Assis – IMESA.

1. Simulador de Controle Automatizado 2. Geração de Vapor 3. PID

CDD: 001.6

Biblioteca da FEMA

JOSÉ TIAGO SILVÉRIO

SIMULADOR DE CONTROLE AUTOMATIZADO DE GERAÇÃO DE VAPOR.

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis (IMESA) e Fundação Educacional do Município de Assis (FEMA), como requisito do Curso de Graduação de Ciência da Computação, analisado pela seguinte comissão examinadora.

Orientador: Profº. Esp. Guilherme de Cleva Farto.

Analisador: Profº. Osmar Aparecido Machado.

Assis

2012

DEDICATÓRIA

Dedico este trabalho a minha família, amigos e todas as pessoas que sempre me apoiaram nos momentos mais difíceis, me dando forças para nunca desistir de meus sonhos.

AGRADECIMENTOS

Primeiramente a Deus, pois somente ele pode ter colocado em minha vida, todas essas pessoas que sempre me ajudaram, e nunca me deixaram desistir de meus sonhos.

A Prof^a. Regina Fumie Eto, pela orientação e confiança que conquistei no tempo que trabalhou na Fundação Educacional do Município de Assis (FEMA).

Ao Prof^o. Esp. Guilherme Cleva Farto, por aceitar dar continuidade ao trabalho que vinha sendo elaborado junto à Prof^a. Regina Fumie Eto.

A todos os professores do curso de Ciência da Computação, que me possibilitaram agregar conhecimento suficiente para conclusão deste trabalho.

A minha família, principalmente a meus pais, que me ensinaram o valor do caráter, e me possibilitaram conquistar tudo o que conquistei. Aos meus tios, pelo incentivo e confiança, e a minha namorada que esteve ao meu lado durante esses anos, me dando forças para conseguir meus objetivos.

E aos amigos, que colaboraram direta e indiretamente na execução deste trabalho.

“Procure ser um homem de valor, em vez de ser um homem de sucesso.”

Albert Einstein

(1879-1955)

RESUMO

Atualmente, a automação é uma área de atuação que tem ganhado muita importância no mundo, devido a esse fator, as indústrias tem investindo cada vez mais em novas tecnologias visando um aumento de seus lucros em um curto espaço tempo. Contudo as despesas com a implantação do sistema e treinamentos específicos tende a aumentar cada vez mais.

Este trabalho apresenta a especificação e implementação de um simulador didático destinado a pequenas, médias, e até mesmo grandes Indústrias, que se possa reduzir suas despesas com treinamentos, sem perder a qualidade. Esse simulador foi desenvolvido tentando aproximar ao máximo a realidade de uma caldeira.

Para que este simulador pudesse ser aproximado da realidade, foram realizadas pesquisas sobre os princípios de geração de vapor, como funcionamento de uma caldeira, as fórmulas de PID entre outros mais minha experiência pessoal de trabalho em uma usina de açúcar e álcool.

Palavras chave: Simulador, geração de vapor, PID

ABSTRACT

Nowadays, automation is a field that has gained much importance in the world, due to this factor, the industry is increasingly investing in new technologies to increase their profits in a short time. However the cost of system deployment and specific training tends to increase more and more.

This study presents the specification and implementation of a simulator for teaching to small, medium, and even large industries, which can reduce your training costs without losing quality. This simulator was developed trying to bring reality to the maximum of a boiler.

For this simulator could be approximated reality, research was conducted on the principles of steam generation, and operation of a boiler, the formulas of PID among other more my personal experience working in a sugar and alcohol.

Keywords: Simulator, steam generation, PID

SUMÁRIO

1. INTRODUÇÃO	11
1.1. JUSTIFICATIVA.....	12
1.2. OBJETIVO.....	13
1.3. CONTEXTUALIZAÇÃO.....	13
2. LEVANTAMENTO TEÓRICO.....	14
2.1. CALDEIRA.....	15
2.2. PID (PROPORCIONAL, INTEGRAL E DERIVADA).....	16
2.3. JAVA.....	21
3. DESENVOLVIMENTO DO TRABALHO.....	23
3.1. METODOLOGIA DE PESQUISA.....	23
3.2. CASO DE USO.....	23
3.3. INTERFACE DO SIMULADOR.....	24
4. CONCLUSÃO.....	37
4.1 TRABALHOS FUTUROS.....	37
5. REFERENCIA BIBLIOGRAFICA.....	38

1. INTRODUÇÃO

A automação está presente em nossas vidas desde a pré-história, com a invenção da roda para transportar materiais, diminuindo o esforço humano. No entanto foi na revolução industrial, no século XXVIII, onde foram desenvolvidas máquinas que pudessem realizar tarefas que eram feitas manualmente. Foi a partir dessa data, que as máquinas e os equipamentos começaram a ser mais valorizados, passando por varias melhorias e modificações, com o intuito de tornar todo tipo de produção mais eficaz, confiável e precisa.

Por volta de 1788, surgiram novas máquinas de sistema hidráulicos e pneumáticos, que foram utilizadas nos setores de produção, para otimizar assim alguns processos de produção. No entanto, com essa otimização da produção, veio um impacto social que durou alguns anos, pois as máquinas deixaram muitos trabalhadores desempregados, onde só permaneceram trabalhando aqueles funcionários que conseguiram se adaptar a nova forma de trabalhar operando as máquinas.

No século XX nos Estados Unidos com Henry Ford iniciou-se a produção em serie, onde a indústria automobilista bateu recordes de produção de carros em menos tempo. Ele utilizava a lógica de relés, que eram gigantescos circuitos lógicos controlados por dispositivos eletromagnéticos. Esse sistema gerou um grande avanço na produção automobilística. Porém o sistema ocupava muito espaço, assim como o trabalho com variáveis analógicas era limitado, sendo a manutenção muito demorada, até mesmo para identificar uma bobina queimada e efetuar o reparo, sem contar que quando necessitavam desenvolver um novo modelo de carro, era necessário sucatear todo o sistema e começar um novo sistema do marco zero.

Em 1947, Willian Shockey, John Barden e Walter Brattain inventaram o transistor, dando origem aos primeiros computadores industriais, que começaram a ser utilizados nas indústrias em 1961. A partir deste momento começaram a surgir novas tecnologias como microcomputadores que podia tomar decisões de controle de máquina tais como: ligá-la, desligá-la, sinalizar defeitos e gerar relatórios operacionais. Dentro desse conceito surgiu a tecnologia que é utilizada até hoje, o Programmable Logic Controller (PLC).

O PLC trouxe grandes vantagens, tais como: economia de espaço, pois toda arquitetura de relés fora substituída pela lógica de software, assim como a possibilidade de integração e controle de variáveis analógicas. Houve uma redução do tempo de manutenção, facilitando o reparo dos problemas diagnosticados. Versatilidade na implantação de modificações já que maioria das alterações era realizada no software.

A utilização de software tem contribuído bastante para a melhora nos processos industriais e, para este trabalho, será abordado o processo de uma usina sucroalcooleira.

Dessa forma para uma indústria sucroalcooleira o processo de fabricação tanto de álcool quanto de açúcar também necessita possuir um bom processo de obtenção do melaço e também do controle da caldeira. Deste modo o controle de processos em uma sala de controle de processos é de fundamental importância para um bom desempenho de produtividade.

O simulador de processos irá auxiliar os novos funcionários na manipulação e controle de caldeiras e de outros equipamentos lotados na usina. A aplicação proposta na monografia será o de processo de controle de caldeiras, tendo em vista que é parte do processo que demanda ainda de bons softwares e hardware.

1.1. JUSTIFICATIVA

As empresas estão em constante busca de novas tecnologias que aumentem sua produtividade em um menor tempo e com uma qualidade maior, com a finalidade de minimizar os custos e conseqüentemente aumentar seus lucros.

Neste sentido e pelo fato de hoje estar próximo às pessoas que atuam nessa área, e a partir destes contatos obter informações que expressam onde o sistema poderia ter uma interface de fácil visualização para o usuário. Atento a estes fatores, é possível criar uma aplicação para simular os processos reais de uma caldeira, otimizando as necessidade de capacitação dos funcionários, evitando assim o mau uso dos equipamentos.

1.2. OBJETIVO

Este trabalho tem como objetivo construir um simulador de controle de processos que supervisione e simule um processo real de geração de vapor, com seus controles e ajustes automáticos de possíveis erros, gerando relatórios em tempo real de todos os acontecimentos.

1.3. CONTEXTUALIZAÇÃO

De acordo com a empresa AALBERG INDUSTRIES (2011), proprietária de um sistema de monitoramento de geração de vapor, onde o sistema tem acesso remoto onde se utiliza como principais informações da “caldeira”, as indicações de variáveis como: status de componentes; histórico de alarmes e gráficos de tendência; telas de acontecimentos dos dispositivos; informações relevantes para o nível gerencial: como por exemplo, o rendimento da caldeira, da produção e consumo de combustível, são coletados para análise.

O simulador de Controle Automatizado de Geração de Vapor, aqui proposto tem como intuito deixar toda a complexidade desse software o mais simples possível, com uma interface de fácil entendimento, sendo que os problemas que mais acontecem, são ocasionados pelo próprio usuário, pois este sistema proposto supervisionará todos os processos ocorridos em tempo real, e atualizando os relatórios o mais rápido possível, para se ter uma melhor eficiência no controle da Geração de Vapor.

Segundo BRUNETTO e BARBOSA (2003), eles têm um projeto que pode supervisionar caldeiras, fornalhas e outros componentes industriais, porém ele é focado apenas em controlar a temperatura, onde se diferencia do projeto proposto, que além de controlar a temperatura, também controlará o nível de água, e todos os componentes básicos para a geração de vapor.

2. LEVANTAMENTO TEÓRICO

A geração de vapor pode ser realizada de diversas e para diversos objetivos, por exemplos as usinas sucroalcooleiras são autossuficientes em energia, pois utilizam como combustível o bagaço de cana. O bagaço de cana é levado até as caldeiras através de esteiras, onde são queimados gerando calor que por sua vez aquece o balão d'água, que como se fosse uma panela de pressão, gera vapor e pode movimentar a moenda e também gerar energia elétrica através do gerador.

No entanto não é possível falar sobre geração de vapor sem dar ênfase ao processo de controle automatizado (Automação industrial), devido às periculosidades de manipulação das altas temperaturas e complexidade provenientes deste processo.

Para uma boa eficiência na produção de vapor é preciso ter:

- Um bom combustível, que no caso das usinas é o “bagaço da cana”, deve ser picado e desfibrado no processo de extração do caldo, tornando-o mais fino e seco possível para facilitar a queima;
- Oxigênio em abundância para elevar a temperatura, que são provenientes dos ventiladores, e os gases gerados na queima são retirados através de exaustores e antes de serem lançados na atmosfera passa por um processo de lavagem dos gases para diminuir a poluição.

E para obter o máximo de eficiência na combustão, é necessário ter um controle preciso sobre as variáveis de processo (bagaço de cana, oxigênio e temperatura), e é neste ponto que entra a automação, com o intuito de aperfeiçoar o processo e diminuir os possíveis erros.

Com a implantação da automação será possível obter um melhor aproveitamento da planta industrial e conseqüentemente o aumento de produção, gerando assim maiores lucros, que é o alvo primordial de toda a empresa.

2.1. CALDEIRA

Quando fala-se em caldeiras, logo pensa-se em uma panela de pressão gigante, onde passa todo o caldo extraído da cana, que é aquecido em alta temperatura, separando assim o açúcar do álcool.

Essa função não é da caldeira, porém para fabricar o açúcar e o álcool, o caldo passa por um processo, que refina o caldo tirando suas impurezas, em seguida é separado dois tipos de mel, o mel pobre que vai ser utilizado na destilaria para produzir o álcool, e o mel rico que vai ser utilizado nas fabricas de açúcar.

No entanto, a função de uma caldeira, é gerar calor, que aquece o balão d'água, "Recipiente onde passa toda agua utilizada para gerar vapor", gerando assim o vapor, que por sua vez será transformado em energia mecânica "vapor movimentando turbina", que irá movimentar a moenda, destilaria, fabrica, gerador de energia elétrica entre outros.

Geralmente a caldeira é utilizada durante a safra, onde o bagaço da cana de açúcar que é o combustível que alimenta a caldeira, está em abundância, e é desativada na entressafra, onde o consumo de energia da usina é reduzido, com a caldeira desativada, é possível fazer os reparos necessários, e só irá ficar ativa novamente algumas semanas antes do início da próxima safra.

A caldeira é acesa manualmente pelos operadores, do mesmo modo que uma churrasqueira, porém coloca-se a lenha no centro da fornalha que em seguida é acesa, de forma que esse processo só é realizado uma vez. Assim que a caldeira for acesa, os operadores apenas acrescentam lenha até que o fogo não corra mais risco de ser apagado.

Após o esse processo, o fogo começa a ser alimentado com o bagaço que está estocado, fazendo com que a moenda comece a funcionar. Como a temperatura sobe, o balão d'água recebe agua, começando, aos poucos, a gerar vapor. O vapor gerado não é aproveitado, portanto é descartado até o momento em que a moenda comece a funcionar. Após o funcionamento da moenda, uma parte do bagaço de cana é levada por esteiras até o dosador que dosará a quantidade necessária para

que não haja desperdício, enquanto que o resto do bagaço é levado por outra esteira até o depósito onde será estocado.

O bagaço que sai da moenda tem uma umidade aproximada de 48%, que passa pelo dosador e vai direto para fornalha, e será queimado suspenso, ou seja, é queimado antes de cair na grelha, pois utiliza um ventilador do tipo espargidor que sopra todo o bagaço pra cima, evitando que o bagaço caia na grelha e seja descartado junto com as cinzas, facilitando a queima, e melhorando assim o tempo de resposta da temperatura.

2.2. PID (PROPORCIONAL, INTEGRAL E DERIVADA)

Para um melhor entendimento sobre PID torna-se preciso possuir o conhecimento de algumas siglas que serão utilizadas ao decorrer do artigo, que são: PV, MV, SV ou SP, ERRO ou DESVIO, AÇÃO DE CONTROLE, AÇÃO DIRETA, AÇÃO REVERSA.

- A sigla PV representa a variável de processo, que a variável que vai ser controlada no decorrer do processo, por exemplo: temperatura, nível de água entre outros.
- A sigla SV ou SP, Setpoint é o valor desejado, o valor determinado para a variável de processo, esse é o valor que vai servir como base para o controle, pois o simulador tem como função tentar aproximar ao máximo o valor atual da MV com o valor determinado do SP.
- A sigla MV, Variável Manipulada é a variável que é utilizada para controlar o processo, essa variável pode ser controlada através de uma posição de uma válvula, (10%, 20%, da válvula aberta), nível de um tanque(10%, 20% do nível de um tanque).
- A sigla Pb é a banda proporcional, é a faixa onde é definido que o modo proporcional deve atuar.

- O termo Erro ou Desvio é a diferença entre SV e PV. PV-SV para ação direta, e SV-PV para ação reversa.
- O termo Ação Reversa significa que se PV aumentar então, MV deverá diminuir e é bastante utilizada em controle de aquecimento.
- O termo Ação Direta significa que se PV diminuir, então MV deverá aumentar e é bem utilizada em controle refrigeração.
- O controle PID calcula um valor de atuação do processo utilizando o valor desejado, e o valor atual da variável de processo. Este cálculo por sua vez, é convertido em um sinal para o controle das válvulas, reles, motor podendo abrir 10% da válvula, rele e até mesmo controlar a RPM de um motor visando o controle estável e preciso do processo.

O PID é a composição de três ações quase intuitivas: P de Proporcional, I de Integral, D de Derivada.

Segundo Novus (p. 3)

No controle Proporcional, o valor de MV é proporcional ao valor do desvio (SV-PV, para ação reversa de controle), ou seja, para desvio zero (Sv=PV), MV=0; à medida que o desvio cresce, MV aumenta até o máximo de 100%. O valor de desvio que provoca MV=100% define a Banda Proporcional (Pb). Com Pb alta, a saída MV só irá assumir um valor alto para corrigir o processo se o desvio for alto. Com Pb baixa, a saída MV assume valores altos de correção para o processo mesmo para pequenos desvios. Em resumo, quanto menor o valor de Pb, mais forte é a ação proporcional de controle.

A figura abaixo ilustra a variação de Pb no processo.

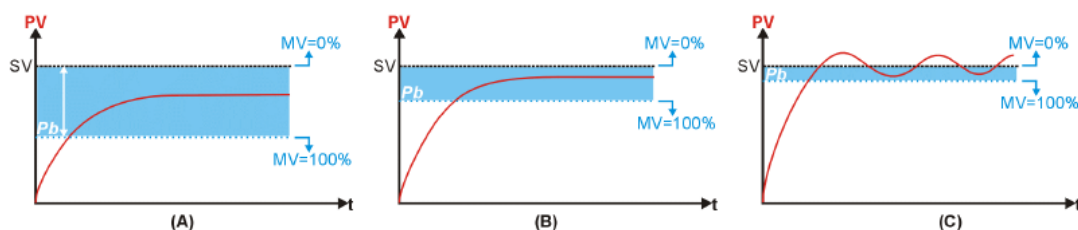


Figura 1 – Efeito da redução de PB no comportamento de PV

Fonte: <http://www.martinezezuim.com.br/arquivos%20pdf/pid%20basico.pdf>

Como fora ilustrado na figura 1(A) com a banda de Pb Grande, o processo consegue se estabilizar, no entanto não vai chegar ao valor desejado. Na figura 1(B) a banda

de Pb foi reduzida deixando assim estabilizar um pouco mais próximo do SetPoint, mas mesmo assim não chega ao valor desejado. E já na figura 1(C) a banda foi reduzida novamente deixando assim a banda bem pequena, porem com uma redução excessiva o processo pode ficar instável ocasionando oscilação. O processo encarregado de gerenciar a banda Pb é chamado de “Sintonia do Controle”.

Os controladores que operam apenas em modo Proporcional trabalham com a MV em um valor constante para que dessa forma possa garantir $PV=SV$.

Segundo Novus (p. 3)

Quando a condição desejada ($PV=SV$) é atingida, o termo proporcional resulta em $MV=0$, ou seja, nenhuma energia é entregue ao processo, o que faz com que volte a surgir desvio. Por causa disto, um controle proporcional puro nunca consegue estabilizar com $PV=SV$.

No entanto não é uma regra trabalhar apenas com Proporcional, pode-se trabalhar com Proporcional e incluindo uma Integral, ou então pode-se trabalhar com Proporcional incluindo uma derivada, e até mesmo trabalhar com Proporcional incluindo Integral e derivada. Porém não é possível trabalhar sem a proporcional, por exemplo: Integral e Derivada, ou apenas Integral ou Derivada.

A imagem abaixo ilustra um processo utilizando o modo Proporcional e outro utilizando o modo Proporcional e Integral.

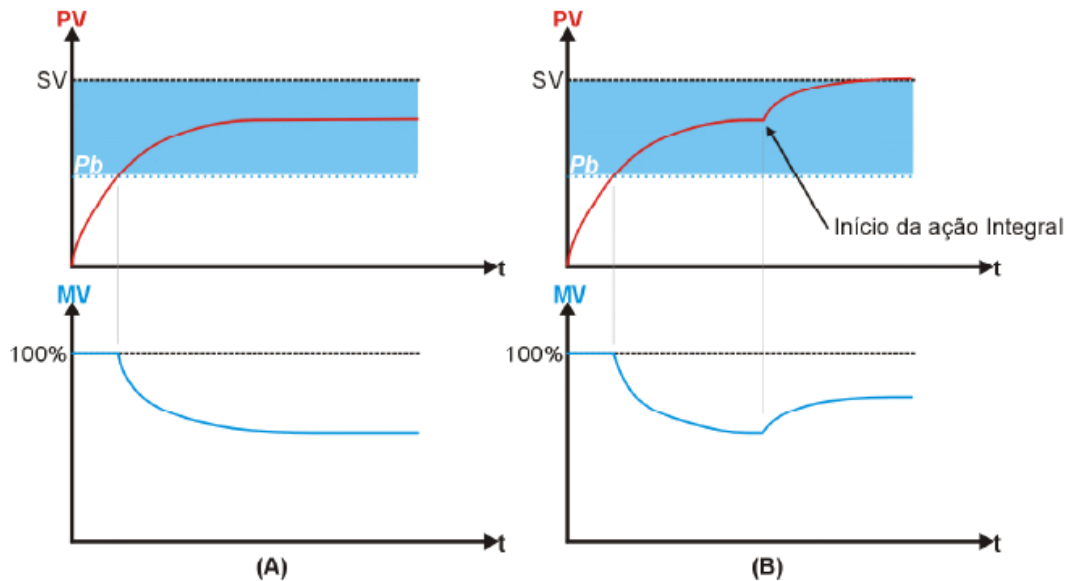


Figura 2 – Efeito da inclusão do controle Integral - PI

Fonte: <http://www.martinezezuim.com.br/arquivos%20pdf/pid%20basico.pdf>

Na figura 2(A) PV e MV atingem um equilíbrio, deixando assim o processo estável, no entanto o processo não deve chegar ao SetPoint (SV), ocasionando assim o chamado Erro em Regime Permanente. Na Figura 2(B), observe o ponto indicado com a seta, início da ação Integral, o valor de MV teve um aumento gradual, dessa forma eliminando o Erro em Regime Permanente, pois com a inclusão da ação Integral, o intuito é alterar a MV progressivamente para que dessa forma possa se eliminar o Erro de PV, isso ocorre até que PV e MV se equilibre novamente, mas agora com $PV=SV$.

Segundo Novus (p.4)

A ação integral funciona da seguinte maneira: A intervalos regulares, a ação integral corrige o valor de MV, somando a esta o valor do desvio $SV-PV$. Este intervalo de atuação se chama Tempo Integral, que pode também ser expresso por seu inverso, chamado Taxa Integral (I_r). O aumento da Taxa Integral - I_r - aumenta a atuação do Integral no controle do processo.

Portanto, a Integral tem como objetivo eliminar o ERRO em Regime Permanente, mas em alguns casos se for utilizada excessivamente, ela pode deixa o processo estável.

Como foi citado anteriormente, não podemos utilizar uma ação Derivada, sem utilizar o modo proporcional, no entanto pode-se utilizar modo Proporcional com a ação Derivada que tem como função de antecipar ou prever o erro.

Segundo Novus (p.4)

O derivativo só atua quando há variação no erro. Se o processo está estável, seu efeito é nulo. Durante perturbações ou na partida do processo, quando o erro esta variando, o derivativo sempre atua no sentido de atenuar as variações, sendo portanto sua principal função melhorar o desempenho do processo durante os transitórios.

A imagem abaixo ilustra uma comparação entre um modo Proporcional (A) e um modo Proporcional integrado com Derivada(B).

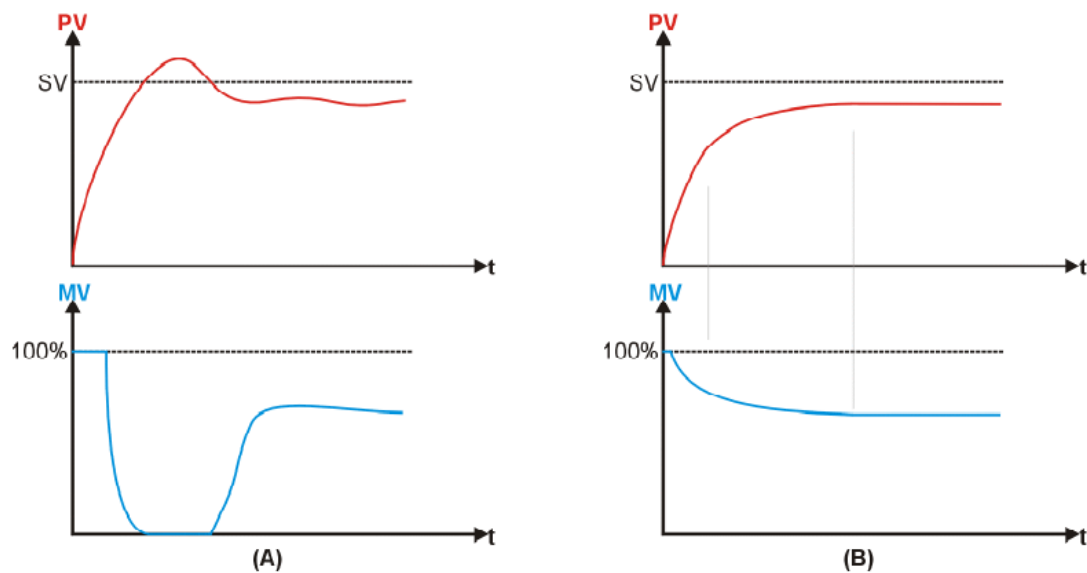


Figura 3 – Comparação de um controle P com um controle PD

Fonte: <http://www.martinezeuim.com.br/arquivos%20pdf/pid%20basico.pdf>

Na figura 3(A), onde ocorre o apenas o modo proporcional, foi definido uma P_b pequena, aconteceu o 'overshoot', que é quando PV ultrapassa SV antes de estabilizar.

Segundo Novus (p. 5)

“Isto ocorre pelo longo tempo em que MV esteve no seu valor máximo e por ter sua redução iniciada muito próximo de SV, quando já é tarde para impedir o overshoot”. Uma possível solução poderia ser o aumento de P_b , no entanto com o aumento de P_b aumentaria por sua vez o Erro em Regime Permanente. Outra possível solução é trabalhar com o conceito de Derivada, como ilustrado na figura 3(B) que reduz de MV se PV esta se elevando rapidamente. No entanto ao prever que PV sofreria uma variação, a ação derivativa pode não apenas reduzir o erro, como pode também eliminar o overshoot e as possíveis oscilações ao decorrer do processo.

Como fora citado anteriormente, pode-se trabalhar com as três técnicas, o modo Proporcional com seu controle básico, adicionando a ação Integral que por sua vez elimina o ERRO em Regime Permanente, e mais a ação derivativa que prevê e reduz as possíveis oscilações diminuindo o overshoot, utilizando assim o conceito de PID, no entanto quando trabalha-se com as três técnicas em conjunto (PID), cria-se então uma dificuldade de ajustar a intensidade de cada um dos termos. Pode-se resolver este problema utilizando a sintonia de PID.

A sintonia de PID pode ser realizada em malha fechada ou malha aberta, onde o processo em malha aberta seria um controle manual, e o processo em malha fechada seria um controle automático.

A maioria dos controladores PID, trabalha em malha fechada, porém para ter resultados mais satisfatórios é necessário trabalhar também em malha aberta, intercalando assim os dois modos de controle para que se possa reduzir o índice de overshoot, garantindo uma estabilização mais rápida do processo.

2.3. JAVA

Para o desenvolvimento do simulador proposto, será utilizado a linguagem de programação Java, que por sua vez tem um grande poder computacional e é totalmente gratuita.

Java é uma linguagem computacional completa, adequada para o desenvolvimento de aplicações baseadas na rede Internet, redes fechadas ou ainda programas stand-alone [CAMPIONE, M; WALRATH, K].

Foi desenvolvida na primeira metade do século 90 no laboratório da Sun Microsystems e tinha como propósito ser mais simples e eficientes que suas predecessoras, para que pudesse desenvolver software com mais qualidade para aparelhos eletrônicos tais como: (micro-ondas, agendas eletrônicas).

A linguagem ficou mais simples e eficiente porém não conseguiu se firmar no comércio de eletrônicos, foi então que os pesquisadores da Sun perceberam a popularização da Internet e resolveram adaptar a linguagem para que pudesse ser utilizada em microcomputadores conectados na World Wide Web (WWW). Após este fato, começaram a surgir novos programas que receberam o nome de Applets, que trafegam pela Web com uma interface gráfica, o que era uma revolução para aquela época. Assim surgiu o primeiro browser compatível com a linguagem, o HotJava responsável por transmitir a interface entre aplicações Java e o sistema operacional. Através desse feito a linguagem Java começou a se popularizar, sendo cada vez mais utilizada.

Java é uma linguagem poderosa em ambientes distribuídos complexos como a rede Internet. Mas sua versatilidade permite ao programador ir além, oferecendo uma poderosa linguagem de programação de uso geral, com recursos suficientes para a construção de uma variedade de aplicativos que podem ou não depender do uso de recursos de conectividade [WUTKA, M.].

Com o conhecimento adquirido através do levantamento teórico, foi possível ter um melhor entendimento de todos os padrões de funcionamento de uma caldeira, priorizando assim o essencial controle de PID, que por sua vez controla vários tipos de processos de forma segura. Desta forma torna-se possível a implementação do simulador proposto, utilizando a linguagem de programação Java, que por sua vez tem um grande poder computacional e é totalmente gratuita.

3. DESENVOLVIMENTO DO TRABALHO

3.1. METODOLOGIA DE PESQUISA

A metodologia inicialmente a ser utilizada no sistema é o de Simulação, onde o software irá simular a temperatura, o nível de água, assim como todos os componentes essenciais para que possa gerar o vapor. Com o intuito de gerar relatórios em tempo real, todos os componentes serão medidos constantemente. No levantamento dos dados coletados e pela emissão do relatório, será possível avaliar e determinar as tomadas de decisão que o simulador poderá executar.

3.2. CASO DE USO

O caso de uso é uma descrição narrativa de uma sequência de eventos, que são representadas por elipses com o nome do caso de uso em seu interior, e por um boneco que representa o ator, que representa quem terá acesso ao sistema, ou as variáveis de processos utilizadas.

Na Figura 4, percebe-se, quatro atores que estão representando as variáveis de processo, onde o ator bagaço terá a função de alimentar a fornalha com bagaço, o ator Fogo terá a função de acender a fornalha, e o ator Ar que alimenta a fornalha com oxigênio elevando a temperatura. Após essas condições serem realizadas, a fornalha é aquecida sendo necessário ter um controle sobre a temperatura.

O ator Água é responsável pela alimentação de água no balão d'água, que necessita de um controle do nível de água, para que não falte água. No entanto a fornalha tem a função de aquecer o balão d'água, que também precisa de um controlador de temperatura por questão de segurança.

O vapor é gerado apenas se a temperatura do balão d'água for superior a 100° C.

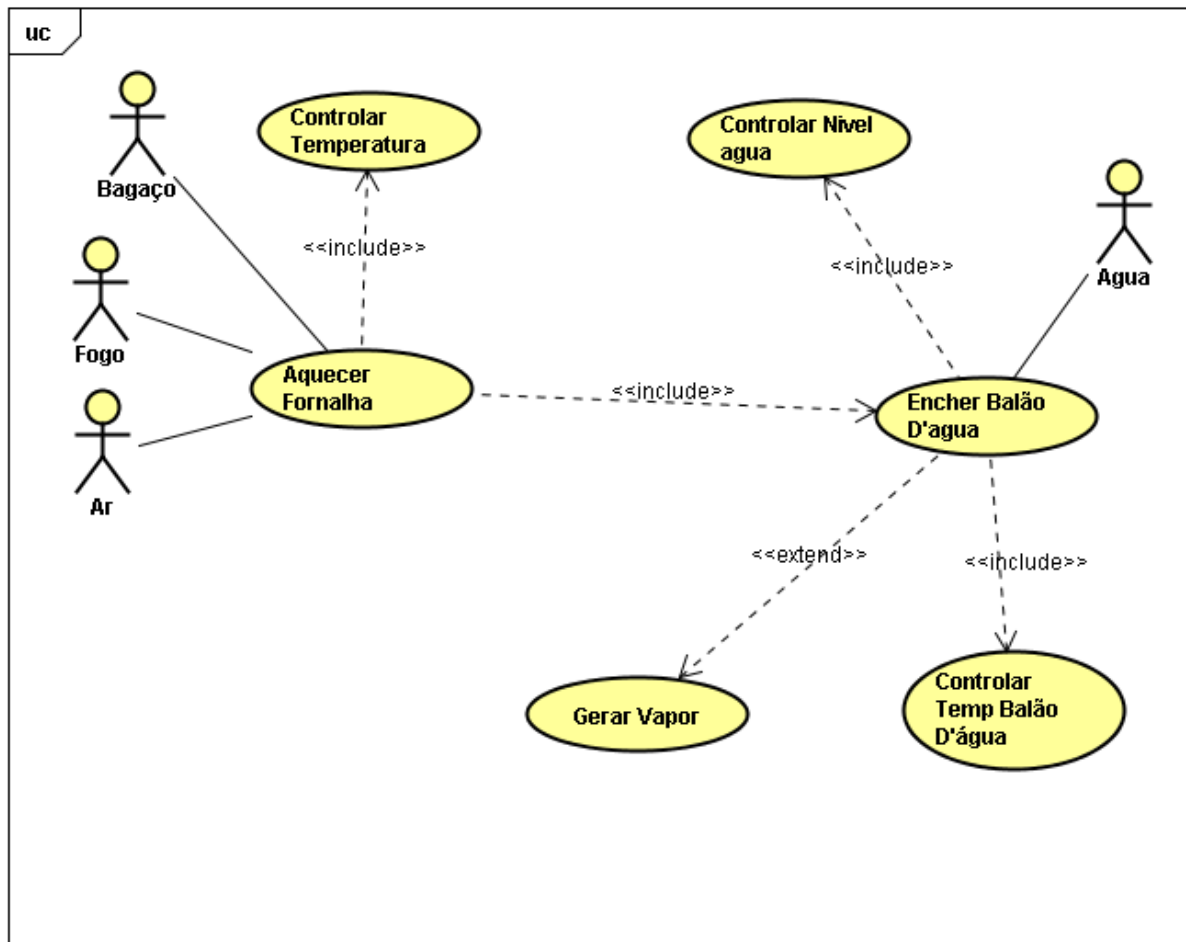


Figura 4 – Caso de Uso Simulador

3.3. INTERFACE DO SIMULADOR

Na figura 5, é possível observar a tela inicial do simulador com todos os processos parados. Para iniciar o processo, deve-se dar um clique sobre cada válvula onde é desejado iniciar o processo, para que seja possível assim determinar, por meio da sintonia de PID o quanto a válvula deve ser aberta.

Para um melhor entendimento sobre o simulador, foram destacados os botões e os campos, onde:

- B1 abre a tela para que se possa alimentar a fornalha com oxigênio.

- B2 abre a tela de alimentação de bagaço.
- B3 abre a tela de geração de vapor.
- B4 abre a tela de sintonia de PID da alimentação de água no balão d'água.
- C1 mostra o percentual de oxigênio que alimentando a fornalha.
- C2 mostra o percentual de bagaço que esta alimentando a fornalha.
- C3 mostra a temperatura que esta a fornalha.
- C4 mostra o percentual atual do balão d'água.
- C5 mostra quantos litros tem no balão d'água.
- C6 mostra a quantidade de vapor gerado.
- C7 mostra a porcentagem de agua que esta alimentando o balão d'água.

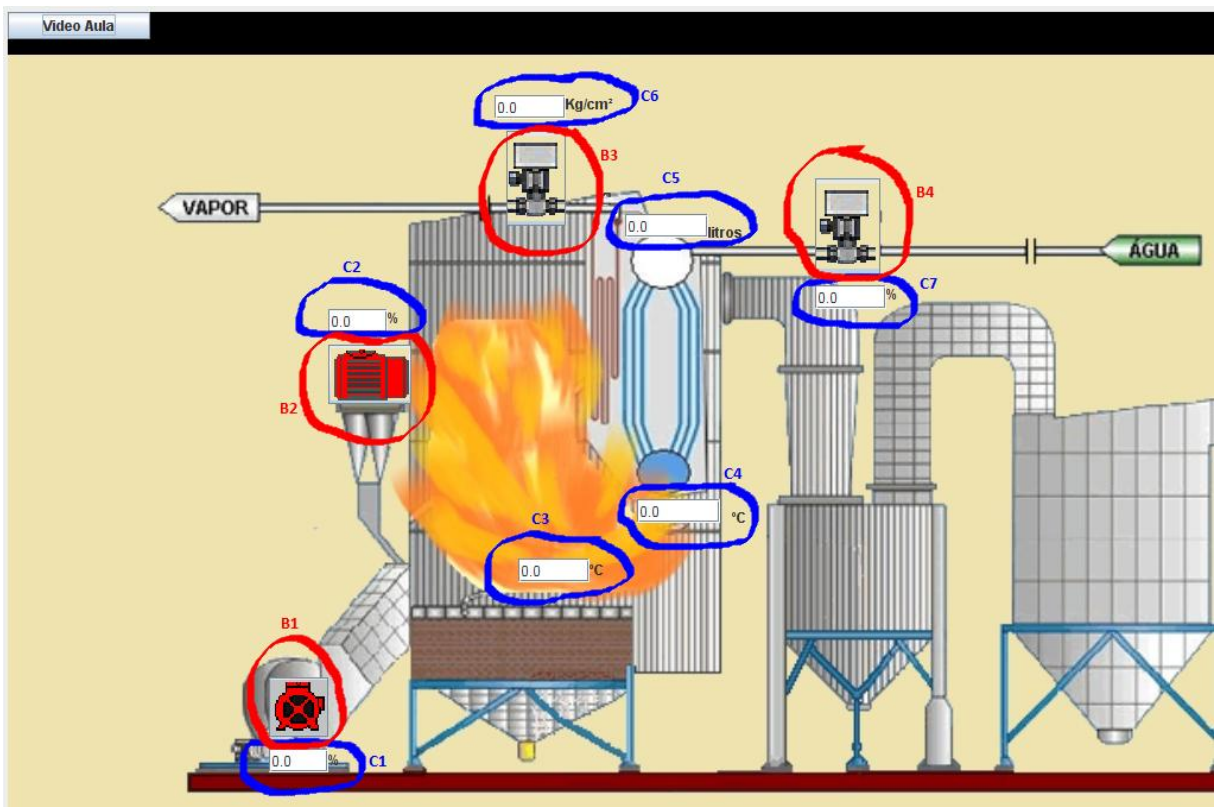


Figura 5 – Tela Principal do simulador

Na figura 6, é possível observar a tela de Controle de PID do Balão D'água, que é acessada pelo botão B4, percebe-se vários campos nessa tela, onde:

- “litros” representa a quantidade de litros que esta no balão d'água, que no momento está com aproximadamente 21.191 litros de água.
- “SP” representa o valor desejado do balão, ou seja esta determinando que o balão irá encher até 60% de sua capacidade.

- “PV” representa a variável de processo atual, ou seja esta mostrando a quantidade atual de água dentro do balão, no caso esta com 35.3%.
- “MV” representa a variável manipulada, ou seja representa a porcentagem que a válvula esta aberta que no caso esta 74.09%.
- Também podemos notar o “nível de água mínimo” e “nível de água máximo” que são utilizados como parte do sistema de segurança da aplicação de forma que quando a PV ficar abaixo de 40%, será enviado um e-mail para o superior ou responsável com uma mensagem avisando que o nível de PV está abaixo do determinado, Caso o valor de PV ultrapassar 60%, também será enviado um e-mail para o superior ou responsável informando que o nível está acima do determinado.
- A aplicação também possui os botões “A” e “M” onde A é automático e o próprio sistema calcula a MV sem deixar que a PV ultrapasse a SP, enquanto que M é manual, onde o usuário assume a responsabilidade e controla PV com valores fixos em MV.
- “Relatório Gráfico”, onde é gerado relatórios em tempo real, em forma de gráfico de linhas.

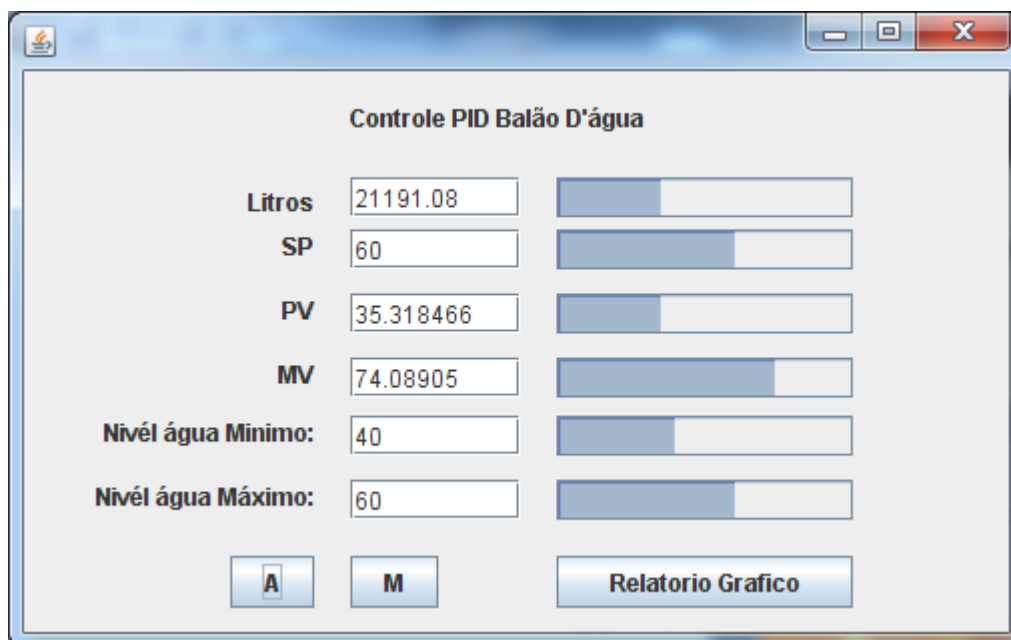


Figura 6 – Tela PID Balão D'água

Na figura 7, é possível observar o controle da sintonia de PID da esteira de bagaço que por sua vez alimenta a fornalha, essa tela pode ser acessada através do botão

B2, pode-se perceber uma redução dos campos nessa tela, pelo motivo de deixar o simulador de fácil entendimento para o usuário, nessa tela o usuário só informa o valor de SP, ou seja, o usuário só informa o percentual de bagaço que irá alimentar a fornalha, e o simulador se encarrega de calcular a temperatura da fornalha.

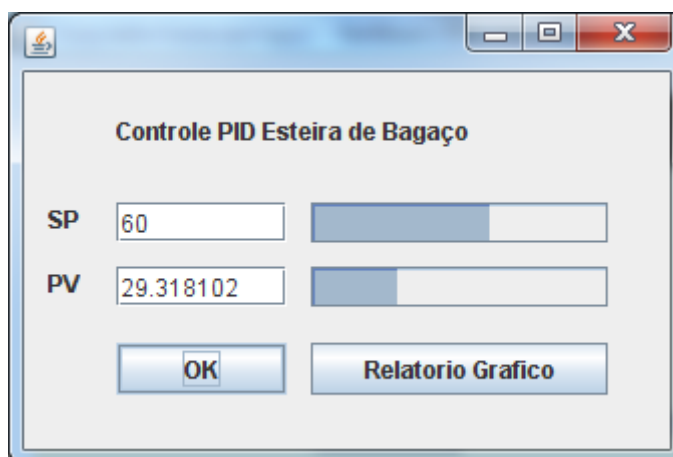


Figura 7 – Tela Controle PID da Esteira de Bagaço

A Figura 8 representa o controle da sintonia de PID do ventilador espargiador, que alimentar a fornalha com oxigênio, essa tela é acessada através do botão B1, onde apenas é preciso informar a porcentagem do valor desejado que o simulador se encarregue de aumentar ou diminuir a taxa de oxigênio dentro da fornalha, ou seja não adianta estar alimentando com 100% de bagaço dentro da fornalha sendo que não há oxigênio suficiente para que o bagaço seja queimado.

Dessa forma, para um melhor aproveitamento do bagaço, deve-se trabalhar com valores próximo na alimentação da fornalha. Por exemplo: se a fornalha for alimentada com 60% de bagaço, deve-se alimentá-la com uma taxa entre 40% a 80% de oxigênio. Já que não adianta possuir muito bagaço e pouco oxigênio, da mesma forma que não adianta ter pouco bagaço e muito oxigênio.

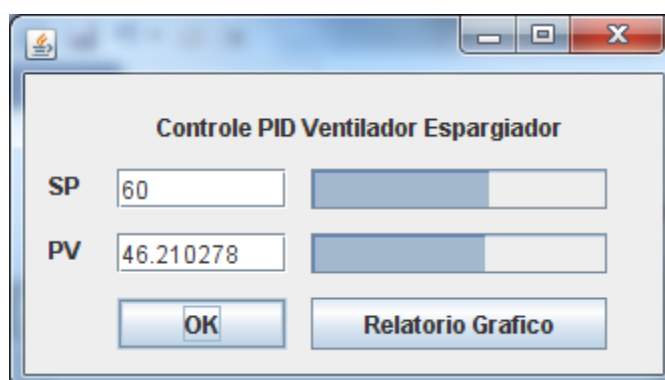


Figura 8 – Controle PID ventilador espargiador

A Figura 9 representa o controle da geração de vapor, que é acessado pelo botão B3, diferente das demais imagens ilustradas, não é possível interferir no vapor que está sendo gerado, pelo menos não por essa função.

A função dessa tela é definir o nível máximo de vapor que, se alcançado, irá enviar um e-mail para o supervisor ou responsável com a mensagem avisando que a caldeira chegou em uma situação extremamente crítica. Por esse motivo, o valor de nível máximo de vapor optado fora 19k/psi para uma caldeira que suporta 21k/psi, pois se ultrapassar esse nível, os operadores possuem algum tempo pra reverter os processos antes que aconteça uma catástrofe ou acidente grave.

Para diminuir o vapor gerado deve-se diminuir a alimentação de bagaço e oxigênio da fornalha fazendo com que a temperatura da fornalha e conseqüentemente a temperatura do balão d'água, diminuam, que por sua vez irá diminuir a quantidade de vapor produzido, reduzindo a pressão.

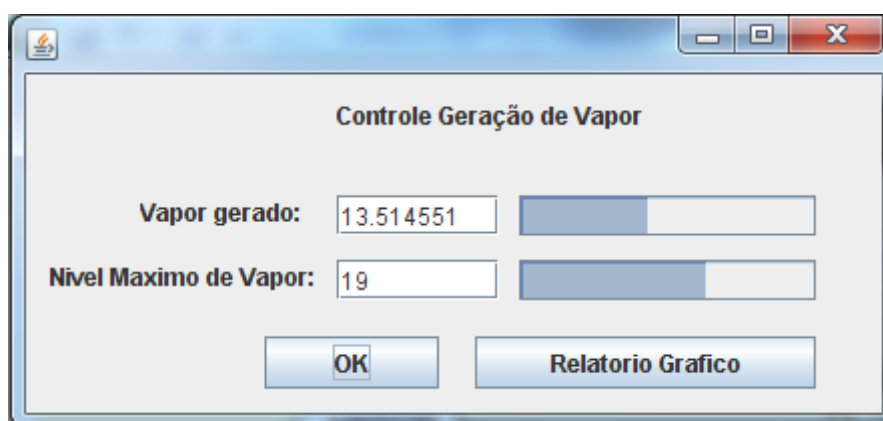


Figura 9 – Tela Controle de geração de vapor

Na Figura 10 é possível observar uma situação onde a caldeira está trabalhando perfeitamente dentro das especificações definidas anteriormente.

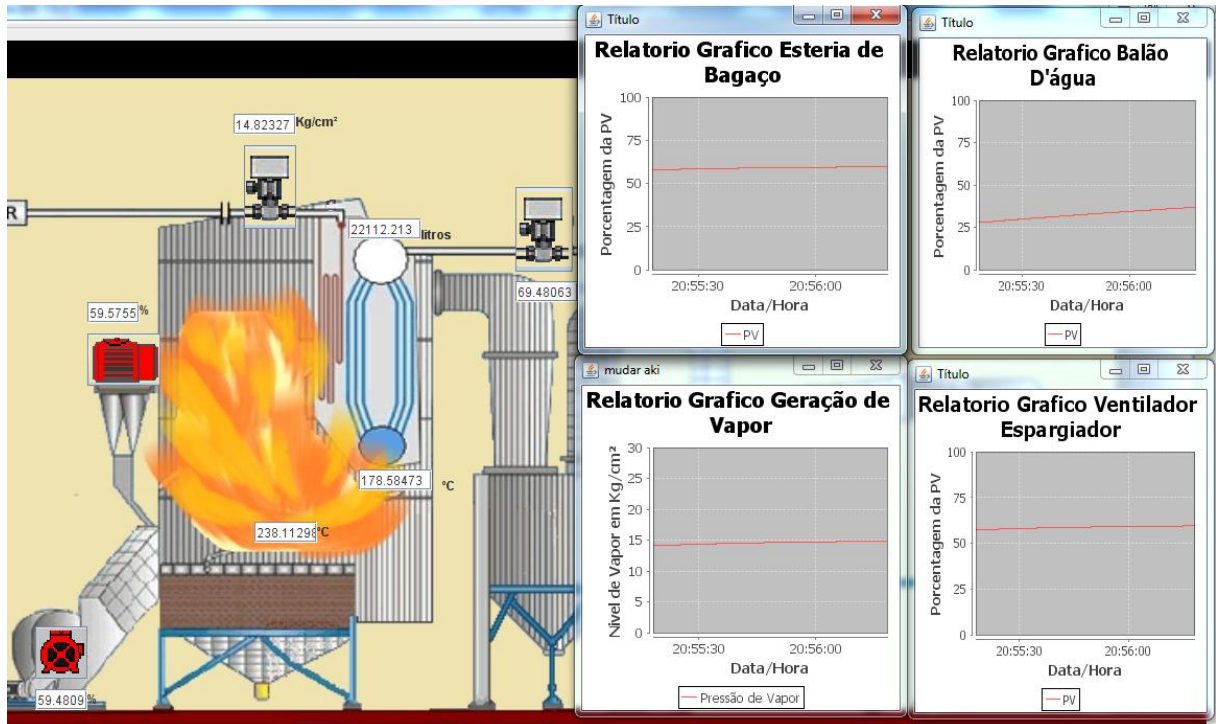


Figura 10 – Tela Principal em execução

A seguir, serão apresentados trechos de código-fonte com suas respectivas descrições, de forma que possam contribuir com novos trabalhos técnicos e acadêmicos, assim como tornar possível a evolução do simulador desenvolvido neste projeto.

No Código 1, é possível observar a criação e a inicialização de uma thread ou linha de execução, que no momento está vazia, porém o objetivo deste pequeno trecho é apenas o de ilustrar como está sendo disparada a thread de calculo.

Código 1 – Criar Threads

01	<code>if (threadCalculo != null) {</code>
02	<code>stopThread = true;</code>
03	<code>try {</code>
04	<code>Thread.sleep(500);</code>
05	<code>} catch (InterruptedException ex) {</code>
06	<code>Logger.getLogger(TelaPIDBalaoDagua.class.getName()).log(Level.SEVERE, null,</code> <code>ex);</code>
07	<code>}</code>
08	<code>}</code>
09	<code>stopThread = false;</code>
10	<code>threadCalculo = new Thread(new Runnable() {</code>
11	<code>@Override</code>
12	<code>public void run() {</code>

13	<code>while (!stopThread) {</code>
14	<code>try {</code>
15	<code>Thread.currentThread().sleep(100);</code>
16	<code>} catch (InterruptedException ex) {</code>
17	<code>Logger.getLogger(TelaPIDBalaoDagua.class.getName()).log(Level.SEVERE, null,</code> <code>ex);</code>
18	<code>}</code>
19	<code>}</code>
20	<code>}</code>
21	<code>});</code>
22	<code>threadCalculo.start();</code>

O Código 2 é responsável por calcular o valor de Proporcional e Integral do balão d'água, onde spBD é definido como o valor desejado do balão, pvBD é definido como a variável de processo do balão, e a mvBD é definido com a posição da válvula, exemplo: a válvula está aberta em 50%. No caso, as variáveis foram definidas em uma classe chamada "Variaveis.java" onde foram definidas como variáveis globais para facilitar o acesso da mesma variável em dois objetos JFrame distintos. O Código 2 está sendo executada através de uma thread.

Código 2 – Calculo PID Balão D'água

01	<code>ThreadCalculo = new Thread (Runnable(){</code>
02	<code>@override</code>
03	<code>public void run() {</code>
04	<code>while (!stopThread && Variaveis.spBD < 101 && Variaveis.spBD !=</code> <code>Float.parseFloat(txtPV.getText())) {</code>
05	<code>Variaveis.spBD = Float.parseFloat(txtSP.getText());</code>
06	<code>Variaveis.pvBD = Float.parseFloat(txtPV.getText());</code>
07	<code>Float KP = 2f; Float TR = 1f;</code>
08	<code>Float m1 = (Variaveis.spBD - Variaveis.pvBD)* KP; /* 2f; // KP</code>
09	<code>Float m2 = (Variaveis.spBD - Variaveis.pvBD) / TR; // TR</code>
10	<code>Variaveis.mvBD = m1 + m2;</code>
11	<code>if (Variaveis.mvBD > 100){</code>
12	<code>Variaveis.mvBD =100f; }</code>

13	Variaveis.pvBD += (Variaveis.mvBD* 0.02f) / 100;
14	Variaveis.qtdlitros = ((Variaveis.pvBD * 60000)/100);
15	txtPV.setText(Float.toString(Variaveis.pvBD));
16	txtMV.setText(Float.toString(Variaveis.mvBD));
17	txtVaLitros.setText(Float.toString(Variaveis.qtdlitros));

O Código 3 é responsável pelo envio de e-mail. Este e-mail é enviado pelo próprio simulador, para o supervisor ou qualquer outra pessoa que for definida, o e-mail só será enviado caso a temperatura exceda o limite, ou o nível de agua esteja abaixo do esperado, ou o nível esteja a cima do limite.

Código 3 – Envio de e-mail

01	<code>import java.util.Properties;</code>
02	<code>import javax.mail.Message;</code>
03	<code>import javax.mail.Session;</code>
04	<code>import javax.mail.Transport;</code>
05	<code>import javax.mail.internet.InternetAddress;</code>
06	<code>import javax.mail.internet.MimeMessage;</code>
07	<code>import javax.mail.Authenticator;</code>
08	<code>import javax.mail.PasswordAuthentication;</code>
09	<code>public class SendMail {</code>
10	<code> private String mailSMTPServer;</code>
11	<code> private String mailSMTPServerPort;</code>
12	<code> /* quando instanciar um Objeto ja sera atribuido o servidor SMTP do GMAIL e a porta usada por ele */</code>
13	<code> SendMail() { //Para o GMAIL</code>
14	<code> mailSMTPServer = "smtp.gmail.com";</code>
15	<code> mailSMTPServerPort = "587";</code>
16	<code> }</code>
17	<code> /* caso queira mudar o servidor e a porta, so enviar para o contrutor os valor como string */</code>

18	SendMail (String mailSMTPServer, String mailSMTPServerPort) { /*Para outro Servidor*/
19	this.mailSMTPServer = mailSMTPServer;
20	this.mailSMTPServerPort = mailSMTPServerPort;
21	}
22	public void sendMail (String from, String to, String subject, String message) {
23	Properties props = new Properties();
24	props.put("mail.transport.protocol", "smtp"); /*define protocolo de envio como SMTP*/
25	props.put("mail.smtp.starttls.enable", "true");
26	props.put("mail.smtp.host", mailSMTPServer); /* server SMTP do GMAIL */
27	props.put("mail.smtp.auth", "true"); //ativa autenticacao
28	props.put("mail.smtp.user", from); /* usuario ou seja, a conta que esta enviando o email (tem que ser do GMAIL) */
29	props.put("mail.debug", "true");
30	props.put("mail.smtp.port", mailSMTPServerPort); //porta
31	props.put("mail.smtp.socketFactory.port", mailSMTPServerPort);
32	/* mesma porta para o socket */
33	props.put("mail.smtp.socketFactory.fallback", "false");
34	//Cria um autenticador que sera usado a seguir
35	SimpleAuth auth = null;
36	auth = new SimpleAuth ("email@gmail.com", "SenhaValida");
37	//Session - objeto que ira realizar a conexão com o servidor
38	/*Como há necessidade de autenticação é criada uma autenticacao que é responsavel por solicitar e retornar o usuário e senha para autenticação */
39	Session session = Session.getDefaultInstance(props, auth);
40	session.setDebug(true); /* Habilita o LOG das ações executadas durante o envio do email */

41	<code>//Objeto que contém a mensagem</code>
42	<code>Message msg = new MimeMessage(session);</code>
43	<code>try {</code>
44	<code>//Setando o destinatário</code>
45	<code>msg.setRecipient(Message.RecipientType.TO, new InternetAddress(to));</code>
46	<code>//Setando a origem do email</code>
47	<code>msg.setFrom(new InternetAddress(from));</code>
48	<code>//Setando o assunto</code>
49	<code>msg.setSubject(subject);</code>
50	<code>//Setando o conteúdo/corpo do email</code>
51	<code>msg.setContent(message, "text/plain");</code>
52	<code>} catch (Exception e) {</code>
53	<code>System.out.println(">> Erro: Completar Mensagem");</code>
54	<code>e.printStackTrace();</code>
55	<code>}</code>
56	<code>//Objeto encarregado de enviar os dados para o email</code>
57	<code>Transport tr;</code>
58	<code>try {</code>
59	<code>tr = session.getTransport("smtp"); /* define smtp para transporte */</code>
60	<code>/* 1 - define o servidor smtp 2 - seu nome de usuario do gmail 3 - sua senha do gmail */</code>
61	<code>tr.connect(mailSMTPServer, "email@gmail.com", "SenhaValida");</code>
62	<code>msg.saveChanges(); // don't forget this</code>
63	<code>//envio da mensagem</code>
64	<code>tr.sendMessage(msg, msg.getAllRecipients());</code>
65	<code>tr.close();</code>
66	<code>} catch (Exception e) {</code>
67	<code>// TODO Auto-generated catch block</code>
68	<code>System.out.println(">> Erro: Envio Mensagem");</code>

69	e.printStackTrace();
70	}
71	}
72	}
73	/* classe que retorna uma autenticao para ser enviada e verificada pelo servidor smtp */
74	class SimpleAuth extends Authenticator {
75	public String username = null;
76	public String password = null;
77	public SimpleAuth(String user, String pwd) {
78	username = user;
79	password = pwd;
80	}
81	protected PasswordAuthentication getPasswordAuthentication() {
82	return new PasswordAuthentication (username,password);
83	}
84	}

O Código 4 é responsável por verificar se a pressão excedeu o limite, em seguida enviar o e-mail ao supervisor ou responsável.

Código 4 – Teste envio de e-mail

01	if (threadCalculo != null) {
02	stopThread = true;
03	try {
04	Thread.sleep(500);
05	} catch (InterruptedException ex) {
06	Logger.getLogger(TelaPIDBalaoDagua.class.getName()).log(Level.SEVERE, null, ex);
07	}
08	}

09	<code>stopThread = false;</code>
10	<code>threadCalculo = new Thread(new Runnable() {</code>
11	<code>int i =0;</code>
12	<code>@Override</code>
13	<code>public void run() {</code>
14	<code>SendMail smPressao = new SendMail();</code>
15	<code>while (!stopThread) {</code>
16	<code>if(Variaveis.pvVapor > Variaveis.VaporNivelMax){</code>
17	<code> i ++;</code>
18	<code> if (i == 1){</code>
19	<code> smPressao.sendMail("email@gmail.com", "email@gmail.com",</code> <code>"ALERTA!", "Pressão Escedeu o limite de: "+ Variaveis.VaporNivelMax);</code>
20	<code> }</code>
21	<code> if(Variaveis.pvVapor < Variaveis.VaporNivelMax){</code>
22	<code> i = 0;</code>
23	<code> }</code>
24	<code> }</code>
25	<code>try {</code>
26	<code> Thread.currentThread().sleep(100);</code>
27	<code> } catch (InterruptedException ex) {</code>
28	<code> Logger.getLogger(TelaPIDBalaoDagua.class.getName()).log(Level.SEVERE,</code> <code> null, ex);</code>
29	<code> }</code>
30	<code> }</code>
31	<code> }</code>
32	<code>});</code>
33	<code>threadCalculo.start();</code>

Durante o desenvolvimento do simulador, houve dificuldade em encontrar formulas para calcular a temperatura da fornalha, a temperatura do balão d'água e a geração de vapor. Essas fórmulas foram substituídas por regras de três simples onde foram parametrizados valores reais de temperatura da fornalha, balão d'água e até mesmo

a geração de vapor. Houve também grande dificuldade no entendimento das fórmulas de PID que foi sanado por um longo tempo de pesquisa sobre o assunto.

4. CONCLUSÃO

Atualmente, a automação é uma área que vem crescendo muito no Brasil e no mundo. Devido a esse fator, Indústrias vêm investindo cada vez mais em novas tecnologias, visando um aumento de seus lucros em menor tempo. Contudo as despesas com a implantação do sistema, treinamentos específicos, tende a aumentar cada vez mais.

O simulador de controle automatizado de geração de vapor é um simulador didático que foi desenvolvido para que empresas de pequeno, médio porte e ate mesmo grandes empresas renomadas, possa usufruir do simulador para que possa dar um diferencial no treinamento de seus operadores de caldeira. No entanto isso é possível devido a tentativa de aproximar ao máximo a realidade das indústrias para o simulador, que por sua vez foi desenvolvido em cima de dados reais de uma caldeira com capacidade máxima de 21 quilos de pressão.

4.1. TRABALHOS FUTUROS

Como o simulador trata-se apenas de controlar o nível do balão d'água, temperatura da fornalha, temperatura do balão d'água e a geração de vapor, como projetos futuros pretende-se o término de implementação desde a saída dos gases da fornalha que passam pelo pré-aquecedor, aquecendo a água que vai direto para o balão d'água, esses gases após saírem do pré-aquecedor, passa pelo lavador de gases onde são retirados a maioria dos metais pesados que são jogados na atmosfera pela chaminé.

5. REFERENCIA BIBLIOGRAFICA

AALBERG INDUSTRIES “Supervisório/Monitoramento”. Disponível em:
<http://www.aalborg-industries.com.br/general.php?ix=72> ACESSADO EM 28/05/2011.

AUTOMAÇÕES “[Histórico da Automação Industrial](http://www.automacoes.com/2008/12/historico-da-automao-industrial.html)”. Disponível em:
<http://www.automacoes.com/2008/12/historico-da-automao-industrial.html>
ACESSADO EM 25/03/2012.

BRUNETTO João Roberto; BARBOSA Luiz Ricardo Ribeiro; “RELATÓRIO TÉCNICO FINAL SISTEMA DE CONTROLE DE TEMPERATURA DE CALDEIRA, 2003”. Disponível em:
<http://www.lami.pucpr.br/engcomp/projetos/finais/SistemaControleTemperaturaCaldeira.pdf>
ACESSADO EM 28/05/2011.

CAMPIONE, M; WALRATH, K, “The Java Tutorial: Object-Oriented Programming for the Internet.[S.l.]: SunSoft Press, 1996”.

CAMPOS, Denis “SINTONIA DE CONTROLE PID”. Disponível em:
<http://www.ebah.com.br/content/ABAAAADNMAC/sintonia-controle-pid> ACESSADO EM 15/06/2012.

INDRUSIAK Leandro. SOARES “LINGUAGEM JAVA 1996”. Disponível em:
http://www2.comp.ufscar.br/~bruno_katekawa/PLP/Java/introjava.pdf ACESSADO EM 22/10/2012

LOURENÇO João. “SINTONIA DE CONTROLADORES P.I.D.”. Disponível em:
<http://alvarestech.com/temp/smar/Pid.pdf> ACESSADO EM 16/06/2012.

MEIRA, Fernando “A História da Automação Industrial” Disponível em:
<http://automaqate.com.br/?p=12> ACESSADO EM 25/03/2012.

NOVUS “CONTROLE PID BÁSICO”. Disponível em:
<http://www.martinezeuim.com.br/arquivos%20pdf/pid%20basico.pdf> ACESSADO EM
12/06/2012.

Projete soluções elétricas. Disponível em:
<http://www.projeteengenharia.com.br/v2/automacao.php> ACESSADO EM 01/10/2012.

RIBEIRO, Paulo Roberto “CURSO SEQUENCIAL DE AUTOMAÇÃO PARA
INDÚSTRIASUCROALCOOLEIRA. APOSTILA DE TREINAMENTO MÓDULO
II – UNAERP”. Disponível em: [http://pt.scribd.com/doc/66607440/48/Automacao-da-area-de-
Geracao-de-Vapor](http://pt.scribd.com/doc/66607440/48/Automacao-da-area-de-Geracao-de-Vapor) ACESSADO EM 17/03/2012.

WUTKA,M. Java: Técnicas Profissionais. [S.I.]: Berkeley, 1997.