


## CONTROLE DE SISTEMA PARA UMA MÁQUINA DE BOLOS

 <https://doi.org/10.56238/arev6n4-334>

**Data de submissão:** 20/11/2024

**Data de publicação:** 20/12/2024

**Luiz Henrique Sousa da Silveira**

Graduando do Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia Campus Manaus Distrito Industrial  
E-mail: luizhenrique.silveira07@gmail.com

**Uri Santos de Souza**

Graduando do Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia Campus Manaus Distrito Industrial  
E-mail: uri.souza@hotmail.com

**Gabriela Fidelis de Oliveira**

Graduanda do Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia Campus Manaus Distrito Industrial  
E-mail: gbrielaaffidelis@gmail.com

**Paulo Victor Vasconcelos Marajó**

Graduando do Curso de Engenharia de Controle e Automação do Instituto Federal de Educação, Ciência e Tecnologia Campus Manaus Distrito Industrial  
E-mail: pauloxd.vasconcelos@gmail.com

**José Cavalcante Lacerda Junior**

Doutor em Ciências do Ambiente e Sustentabilidade na Amazônia. Professor do Instituto Federal de Educação, Ciência e Tecnologia Campus Manaus Distrito Industrial  
E-mail: jose.cavalcante@ifam.edu.br

### RESUMO

Este artigo estuda a aplicação de Redes de Petri e Autômatos Finitos para a otimização de sistemas em uma máquina de bolos. As Redes de Petri são empregadas para modelar e visualizar as interações dinâmicas entre os diversos componentes do sistema, fornecendo uma representação gráfica robusta e detalhada dos fluxos de trabalho e processos. Por outro lado, os Autômatos Finitos são utilizados para capturar e representar os estados discretos e as transições do sistema, permitindo uma análise precisa e rigorosa dos comportamentos esperados e possíveis. A integração dessas duas abordagens proporciona uma metodologia abrangente para a otimização operacional da máquina de bolos. As Redes de Petri permitem identificar e analisar pontos críticos e gargalos no sistema, enquanto os Autômatos Finitos facilitam a definição clara dos estados e a gestão das transições entre eles. Essa combinação possibilita uma análise detalhada das interações sistêmicas, permitindo a implementação de melhorias estruturadas e eficientes que resultam em um desempenho otimizado. Além disso, a aplicação conjunta dessas técnicas proporciona uma plataforma flexível e adaptável para a simulação e validação de diferentes cenários operacionais, garantindo que as mudanças implementadas sejam efetivas e sustentáveis. Em resumo, este artigo demonstra como a junção entre Redes de Petri e Autômatos Finitos pode ser explorada para alcançar uma otimização significativa na operação de

máquinas de bolos, destacando a relevância e a aplicabilidade dessas ferramentas na engenharia de sistemas complexos.

**Palavras-chave:** Redes de Petri, Autômatos Finitos, Otimização de sistemas, Modelagem Dinâmica, Engenharia de sistemas.

## **1 INTRODUÇÃO**

A indústria alimentícia experimenta uma transformação notável com a chegada da automação e de tecnologias de ponta. Em particular, no setor de confeitaria, as máquinas automatizadas têm um papel fundamental na uniformização e eficácia dos processos produtivos. Este texto pretende investigar a melhoria do processo operacional em equipamentos destinados à fabricação de bolos, um setor crucial da indústria de alimentos.

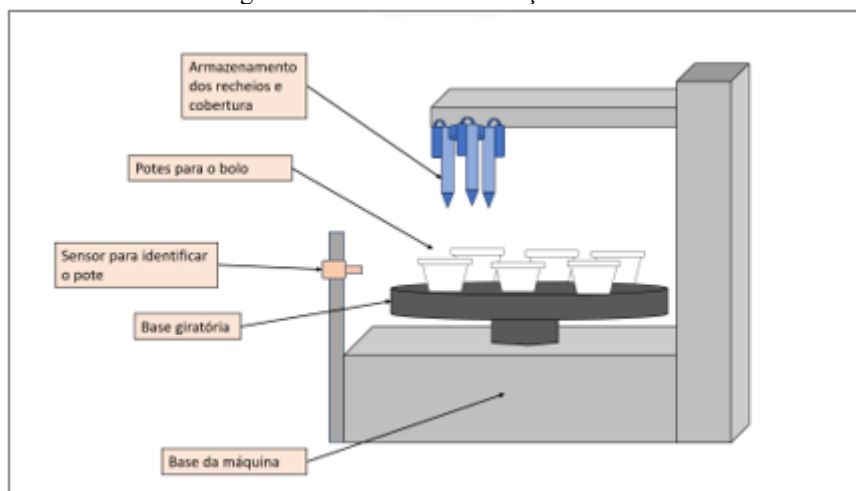
Com o aumento da procura por bolos de alta qualidade e variedades personalizadas, torna-se essencial melhorar a eficácia do processo operacional. Esta busca se concentra no equilíbrio delicado entre a preservação da qualidade artesanal e a maximização dos recursos. Neste cenário, vamos analisar as táticas, tecnologias e melhores práticas que podem ser aplicadas para melhorar a produção de bolos em quantidade, preservando a integridade sensorial e gustativa que os consumidores procuram.

Neste artigo, discutiremos as várias opções para aprimorar um processo, que vai desde a escolha meticulosa dos ingredientes até a automação física inteligente do processo, enfatizando estudos de caso, tendências emergentes e perspectivas futuras. Ao agir dessa maneira, nosso objetivo não é somente aprimorar a eficácia da produção, mas também elevar a experiência oferecida pela interação entre o usuário e a máquina.

## **2 DEFINIÇÃO DO PROBLEMA E PROPOSTA DE OTIMIZAÇÃO DO PROCESSO**

Neste sistema de produção de bolos automatizados, é utilizado um estoque de massa de bolo pronta, armazenado para ser utilizado conforme necessário. O operador alimenta o processo adicionando a massa aos potes, que são então colocados em uma plataforma giratória da máquina. O processo é composto por seis passos distintos: inicialmente, a alimentação do processo com os potes; em seguida, a ativação do sistema; o cliente escolhe a opção de recheio e se deseja cobertura; a adição do recheio é feita conforme a escolha do cliente; a adição ou não da cobertura também é realizada conforme a preferência do cliente; finalmente, o cliente pode retirar seu bolo. Como podemos observar na Figura 1.

Figura 1: Processo de confecção de bolos



Fonte: Autores, 2024

Os propósitos a serem alcançados por meio da otimização do processo podem ser sintetizados com o intuito de diminuir o tempo total de operação, economizar materiais e aprimorar a distribuição de recheios e cobertura ao longo do procedimento de confecção do bolo. Essas metas visam não apenas aumentar a eficiência do processo, mas também contribuir para a eficácia global da produção, resultando em benefícios significativos em termos de economia de tempo, redução de desperdícios e aprimoramento da qualidade do produto final a ser entregue ao cliente.

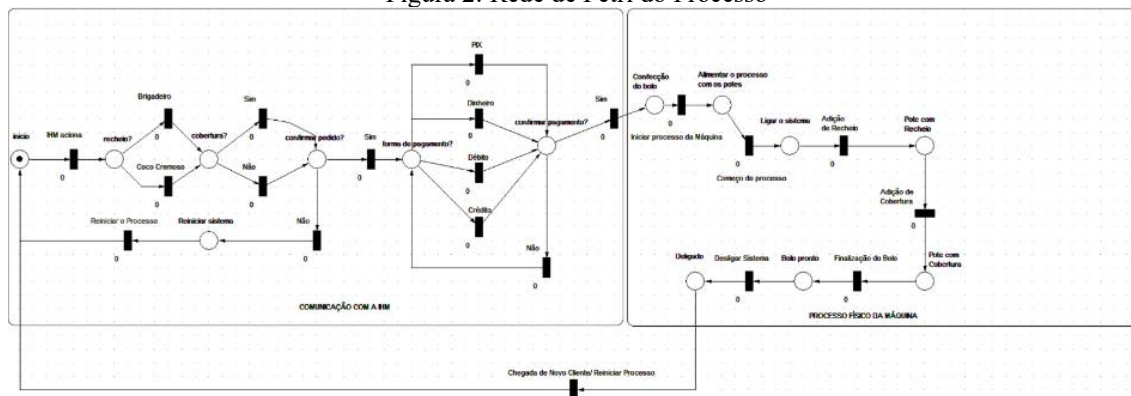
## 2.1 MODELAGENS DO SISTEMA

Paradigmas computacionais como Redes de Petri e Autômatos Finitos têm desempenhado um papel crucial na modelagem e otimização de sistemas complexos, oferecendo ferramentas poderosas para analisar e aprimorar o desempenho de processos em diversas áreas. Redes de Petri, introduzidas por Carl Adam Petri na década de 1960, representam graficamente a dinâmica de sistemas concorrentes, proporcionando uma visão visual e formal para a compreensão de interações entre componentes. Pesquisas recentes, como as de (Murata *et al.*, 2016), destacam a aplicação de Redes de Petri na modelagem de sistemas distribuídos, demonstrando sua eficácia na análise de paralelismo e concorrência.

Por sua vez, Autômatos Finitos emergem como uma abstração fundamental para a representação de comportamentos sequenciais em sistemas computacionais. Trabalhos como o de (Hopcroft *et al.*, 2006) destacam a versatilidade dos Autômatos Finitos em modelar linguagens formais, tornando-os instrumentos valiosos para a otimização de processos com lógicas de controle definidas. Neste artigo, exploraremos as contribuições desses modelos, discutindo casos de estudo relevantes e evidenciando a relevância de Redes de Petri e Autômatos Finitos na otimização de sistemas modernos.

O programa utilizado para realizar a modelagem e representação da rede de Petri do processo em questão foi o Visual Object Net ++, através dele foi possível criar um ambiente controlado onde se pode visualizar todas as etapas do processo, de forma detalhada.

Figura 2: Rede de Petri do Processo



Fonte: Autores, 2024

Através da rede de Petri, podemos observar as etapas do processo, incluindo seus ramos iniciais e os derivados, demonstrando visualmente como seria a atuação da máquina em tempo real. Dessa forma, o artigo intenciona demonstrar soluções que buscam facilitar a gestão do sistema, tornando-o eficiente e automatizado, no processo de pedidos e confecção de bolos. Ele abrange desde a escolha de recheios e coberturas até a confirmação do pedido, forma de pagamento e finalização do bolo. O sistema visa otimizar o fluxo de trabalho, minimizar erros e proporcionar uma experiência de atendimento ao cliente mais eficaz.

### 3 METODOLOGIA

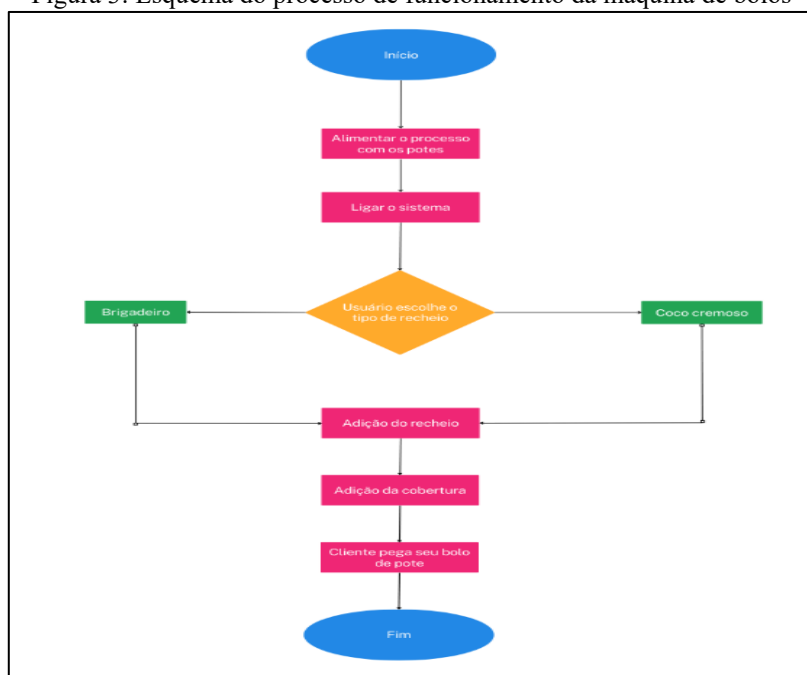
#### 3.1 CARACTERÍSTICAS DO FUNCIONAMENTO DO PROCESSO

O sistema modelado pela rede de Petri apresentada é destinado à gestão de pedidos e confecção de bolos. Ele compreende diferentes estados (lugares) e transições que representam a evolução do processo, desde o início até a entrega do bolo. A interação entre os lugares e transições é guiada por arcos, refletindo as condições e eventos que desencadeiam mudanças no sistema. Comparado com abordagens tradicionais ou sistemas manuais para a gestão de pedidos de bolos, a modelagem usando uma rede de Petri oferece benefícios significativos. Em comparação com sistemas lineares ou sequenciais, ao utilizar as redes de Petri se torna possível representar melhor a concorrência e a simultaneidade de eventos, proporcionando uma visão mais abrangente do comportamento do sistema. Além disso, a modelagem oferece uma representação visual clara do processo, facilitando a compreensão e a análise.

A diferencial chave desta solução é sua capacidade de modelar complexidades inerentes ao processo de confecção de bolos, considerando variáveis como escolhas de recheios, coberturas e formas de pagamento. A flexibilidade da rede de Petri permite adaptar-se facilmente a mudanças nos requisitos do processo, tornando-a uma escolha escalável e dinâmica. Além disso, a representação visual intuitiva da rede de Petri facilita a identificação de gargalos e a otimização contínua do processo.

#### 4 DESENVOLVIMENTO DO SISTEMA DA MÁQUINA DE BOLOS

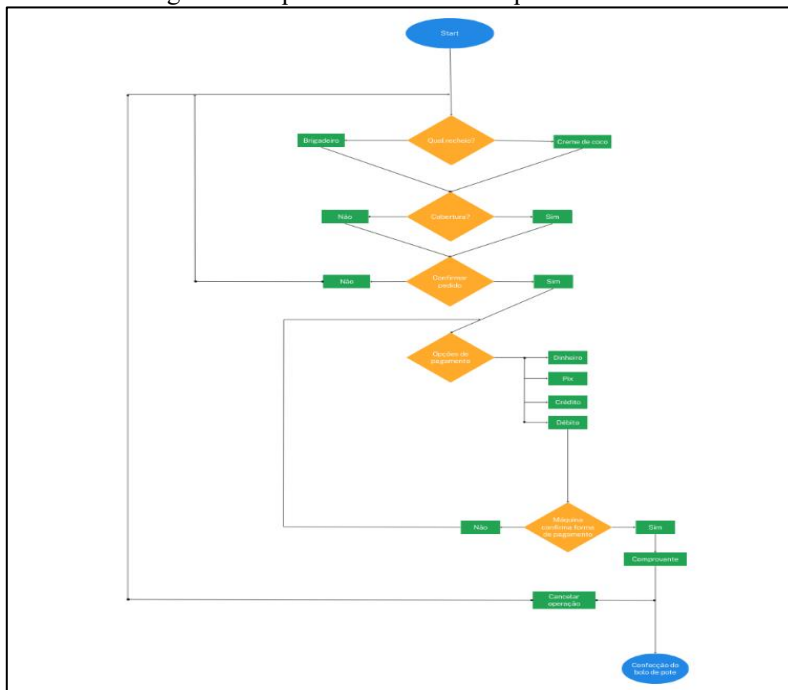
Figura 3: Esquema do processo de funcionamento da máquina de bolos



Fonte: Autores, 2024

Como pode ser visto na figura acima, temos uma representação de como será executado todo o processo de confecção dos bolos, desde o processo de alimentação quanto o de preparação e posteriormente o seu recebimento. Utilizamos a massa de bolo pronta, que ficará armazenada e conforme a necessidade de alimentar o processo, o operador vai colocando a massa dentro do pote e coloca na plataforma giratória da máquina. O processo faz 1 bolo a cada 2min, o processo é dividido em 6 passos, que consistem em alimentar o processo com os potes, ligar o sistema, o cliente escolhe sua opção de recheio e se ele quer cobertura, em seguida é feita a adição do recheio conforme a escolha do cliente, depois é feita a adição ou não da cobertura conforme a escolha do cliente, por fim o cliente pode retirar seu bolo.

Figura 4. Esquema da IHM da máquina de bolos



Fonte: Imagem do autor

A estrutura de casos é essencial para a eficiência de um sistema pois permite a organização e categorização de informações de maneira lógica e acessível. Isso facilita a análise e resolução de problemas complexos de forma sistemática, garantindo decisões mais rápidas e precisas baseadas em critérios pré-definidos para cada situação específica. Observando a imagem representada na figura 4, podemos obter de forma mais detalhada a estrutura de casos do sistema desenvolvido, com suas condições de operação, demonstrando assim de forma mais visual e clara a rota obtida pelo sistema ao completar cada uma das operações ou passos durante o processo inicial ao final.

#### 4.1 ESTRUTURAS DO SISTEMA DE CONFECCÃO DE BOLOS EM REDES DE PETRI:

A elaboração formal do modelo de Rede de Petri para o sistema de processamento da máquina de confecção de bolos de pote foi meticulosamente construída mediante a aplicação de diversos parâmetros específicos. Esses parâmetros foram cuidadosamente selecionados com a finalidade de segmentar o processo, possibilitando a obtenção do diagrama de funcionamento correspondente. Tais parâmetros encontram-se definidos da seguinte maneira:

Quadro 1: Definição de estruturas e parâmetros da rede de petri

Estruturas	Parâmetros
Conjunto de Lugares (P)	$P = \{ \text{"Início"}, \text{"Recheio?"}, \text{"Cobertura?"}, \text{"Confirmar pedido?"}, \text{"reiniciar sistema"}, \text{"Forma de pagamento?"}, \text{"Confirmar pagamento?"}, \text{"Confecção do bolo"}, \text{"alimentar processo"}, \text{"ligar sistema"}, \text{"pote com recheio"}, \text{"pote com cobertura"}, \text{"bolo pronto"}, \text{"desligado"} \}$
Conjunto de Transições (T)	$T = \{ \text{"IHM aciona"}, \text{"brigadeiro"}, \text{"coco"}, \text{"sim"}, \text{"não"}, \text{"pix"}, \text{"dinheiro"}, \text{"debito"}, \text{"crédito"}, \text{"não"}, \text{"sim"}, \text{"iniciar"}, \text{"começar"}, \text{"adição recheio"}, \text{"adição da cobertura"}, \text{"finalizar bolo"}, \text{"desligar sistema"}, \text{"novo cliente"}, \text{"reiniciar sistema"} \}$
Conjunto de Arcos	$A = \{ (\text{"início"}, \text{"IHM aciona"}), (\text{"IHM aciona"}, \text{"Recheio?"}), (\text{"Recheio?"}, \text{"brigadeiro"}), (\text{"Recheio?"}, \text{"coco"}), \dots \}$
Marcas Iniciais	$M(\text{"início"}) = 1$
Função de Incidência	$I : (P \times T) \cup (T \times P) \rightarrow N$ $I(\text{"início"}, \text{"IHM aciona"}) = 1$ $I(\text{"IHM aciona"}, \text{"Recheio?"}) = 1$

## 4.2 INTERPRETAÇÕES DA REDE DE PETRI

**Marcas nos Lugares:** As marcas nos lugares representam o estado atual do sistema.

**Transições:** As transições indicam eventos ou ações que podem ocorrer quando os lugares de entrada têm marcas suficientes.

**Arcos:** Os arcos conectam lugares e transições, indicando a influência de um sobre o outro.

## 4.3 EXECUÇÕES NA REDE DE PETRI

Para a efetiva implementação do sistema, optou-se por delinear uma estrutura abrangente de casos, abarcando assim as diversas etapas inerentes ao processo em questão. Estas etapas foram minuciosamente definidas, visando uma cobertura completa e eficiente do desenvolvimento. Dessa maneira, estabeleceram-se categorias específicas que compõem a estrutura, englobando elementos cruciais para a execução fluida e bem-sucedida do sistema proposto. Sendo definidas em:

Quadro 2: Definição das etapas que podem acontecer em cada caso.

Casos	Etapas
Caso de Sucesso	- Marca inicial em "início". - Transição "IHM aciona" é disparada. - Marca é movida para "Recheio?". - Continua o fluxo até chegar ao estado "bolo pronto".
Caso de Falha	- Tentativa de realizar uma ação fora do fluxo padrão. - Transição correspondente não é disparada. - Leva a um estado de bloqueio.
Caso de Reinício	- Transição "reiniciar sistema" é disparada. - Leva o sistema de volta ao estado inicial. - Permite simular o comportamento do sistema de forma visual e compreensível.

## 4.4 ANÁLISE E CLASSIFICAÇÃO DA REDE DE PETRI DESENVOLVIDA:

A estrutura da Rede de Petri desenvolvida é composta por lugares, transições e arcos, oferecendo a capacidade de modelar o comportamento do sistema em questão. No que tange às propriedades das posições (lugares), é relevante considerar aspectos como a presença ou ausência de



marcações, podendo um lugar ser marcado (contendo uma marca) ou não marcado. Inicialmente, destaca-se que o lugar "início" é marcado. Quanto à temporalidade das marcações, não há informações explícitas que indicam se estas são temporárias ou permanentes. Essa característica dependerá das regras específicas do sistema modelado, as quais não foram detalhadas na descrição.

No âmbito das transições, é crucial compreender se estas estão habilitadas ou não. Uma transição é considerada habilitada quando todos os lugares de entrada estão marcados. Como exemplo, a transição "IHM aciona" está habilitada apenas quando o lugar "início" está marcado. Adicionalmente, ao analisar as propriedades das transições, observa-se que, com base na descrição fornecida, as transições parecem seguir um padrão determinístico. Isso significa que, para uma configuração dada, existe uma única transição possível.

No contexto de propriedades globais, a conservação de marcas é mencionada como uma característica comum em Redes de Petri. No entanto, na descrição há informações específicas sobre como ocorre a Rede de Petri, que é caracterizada como um circuito direcionado, ordinário, puro, máquina de estado. Esta propriedade particular da rede, conhecida por ser conservadora, reforça que a soma total de marcas em todos os lugares permanece constante durante a execução, mantendo-se segura e limitada para  $k=1$ . Além disso, a rede de Petri apresenta conflitos do tipo confusão, aspecto inerente a sua natureza de conservação. Em relação a observações adicionais, é ressaltado que o modelo da Rede de Petri apresentado é relativamente simples e pode ser expandido ou refinado conforme necessário, de acordo com os detalhes específicos do sistema que está sendo modelado.

#### 4.5 LÓGICAS DA REDE DE PETRI PARA O SISTEMA DE CONFEÇÃO DE BOLOS:

Na figura 2, podemos visualizar uma rede de Petri, projetada especificamente para a resolução do problema em questão e ter uma base de como dar-se o processo da confecção e interação da máquina de bolos com o usuário. A estrutura da rede de Petri apresentada é composta por diversos pontos, cada um representando um estado específico do sistema. Inicialmente, temos o ponto p1, que corresponde ao estado "início". Em seguida, a rede segue para os pontos p2 e p3, representando as escolhas sobre o "Recheio?" e a "Cobertura?" do bolo, respectivamente. O processo avança para p4, onde o usuário deve "Confirmar pedido?". Em caso de necessidade de reiniciar o sistema, o ponto p5 é acionado. A escolha da "Forma de pagamento?" é representada por p6, seguida pelas transições para t8, t9, t10 e t11, que correspondem às opções de pagamento via pix, dinheiro, débito ou crédito, respectivamente. O processo segue para p7, onde o usuário deve "Confirmar pagamento?". O ponto p8 marca a fase de "Confeção do bolo", alimentada pelo processo em p9 e iniciada com a ligação do sistema em p10. Os

estados intermediários incluem a presença dos potes com recheio (p11) e cobertura (p12), culminando no bolo pronto em p13. Após a conclusão, o sistema é desligado em p14.

As transições t14 a t17 indicam o fluxo de adição de recheio e cobertura, levando à finalização do bolo em t18. O sistema é desligado em t19, retornando ao ponto inicial p1. O ponto p20, representando a chegada de um novo cliente, é conectado a p1, enquanto p5 e t21 possibilitam a reinicialização do sistema.

Essa estrutura é visualizada através dos arcos que conectam os diferentes pontos e transições. Por exemplo, o arco de p1 para t1 indica que a IHM aciona a transição t1 quando o sistema está no estado "início". Da mesma forma, os arcos subsequentes refletem as relações direcionadas entre lugares e transições, definindo como os estados do sistema são influenciados pelas transições e vice-versa. Essa representação na rede de Petri oferece uma visão clara das dinâmicas do sistema, evidenciando as interações entre os diferentes elementos envolvidos.

#### 4.6 ESTRUTURAS DO SISTEMA DE CONFECCÃO DE BOLOS EM AUTÔMATOS FINITOS:

Para modelar o sistema utilizando um autômato discreto, optamos por um autômato finito determinístico (AFD) ou não determinístico (AFN), dependendo da complexidade do sistema. O autômato escolhido para o desenvolvimento do projeto foi o finito não determinístico (AFN), sendo a estrutura dele definida como:

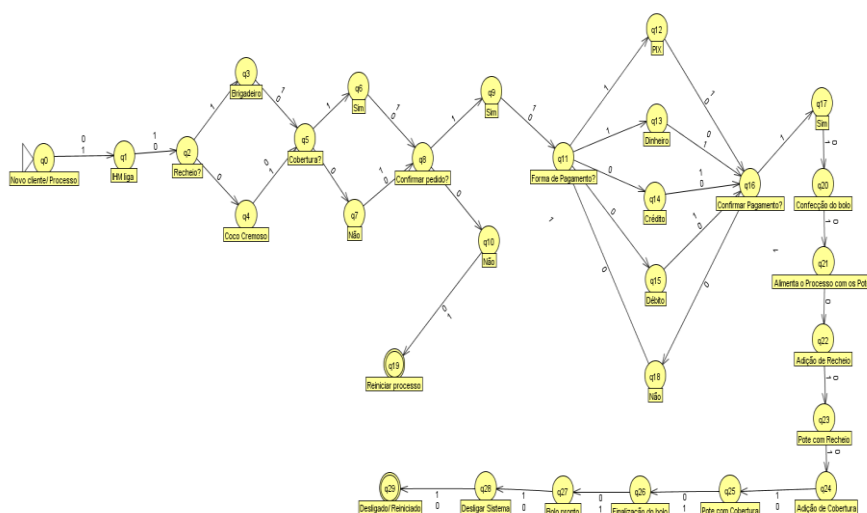
Quadro 3: Estrutura e definições para autômatos finitos.

Conjunto de Estados (Q)	q0, q1, q2, q3, q4, q5, q6, q7, q8, q9, q10, q11, q12, q13, q14, q15, q16, q17, q18, q19, q20, q21, q22, q23, q24, q25, q26, q27, q28, q29
Alfabeto de Entrada ( $\Sigma$ )	0, 1
Conjunto de Estados Iniciais	q0 (Novo cliente/Processo)
Conjunto de Estados Finais	q19 (Reiniciar processo), q29 (Desligado/Reiniciado)

O autômato modela o fluxo do sistema desde o estado inicial ("início") até os estados finais desejados ("bolo pronto" e "desligado"). Ele reflete a dinâmica do processo de confecção de bolos, cobrindo as etapas de escolha de recheios, coberturas, confirmação do pedido, escolha da forma de pagamento, e finalização do bolo. Para as suas simulações foi escolhido o programa JFLAP, pois o uso dele oferece vantagens devido à sua interface gráfica intuitiva e recursos que facilitam a construção, visualização e simulação de autômatos.

O JFLAP simplifica o processo de design de AFDs, permitindo aos usuários modelarem, editar e analisar facilmente as transições e estados do autômato. Além disso, o software fornece funcionalidades de validação e simulação, auxiliando na detecção de erros e no entendimento do comportamento do AFD. Essas características fazem do JFLAP uma ferramenta eficaz para o ensino e aprendizado de teoria da computação, contribuindo para uma compreensão prática e visual dos conceitos relacionados a autômatos finitos determinísticos.

Figura 5. Esquema do Autômato (AFN)



Fonte: Imagem do autor

A figura 4 representa o modelo obtido através do programa JFLAP, após o seu processo de confecção e simulação.

#### 4.8 ANÁLISES DA LINGUAGEM INTERPRETADA PELO AUTÔMATO

AFN, ou Autômato Finito Não-Determinístico, é um modelo teórico utilizado em teoria da computação para representar sistemas computacionais e processos de reconhecimento de linguagens formais. Ao contrário dos autômatos finitos determinísticos (AFDs), nos quais a transição de um estado para outro é determinística, em um AFN, um estado pode ter várias transições possíveis para diferentes entradas ou para a mesma entrada. No projeto desenvolvido, uma sequência possível poderia ser "IHM aciona, sim, brigadeiro, confirmar pedido, pix, bolo pronto, desligado", essa sequência de eventos levaria o sistema pelos estados relevantes do autômato. Para melhor exemplificação dos casos, foram apresentadas algumas situações com diferentes strings.

Quadro 4: Exemplos de combinações valida e rejeitada do sistema em AFN.

Strings	Etapas
String Aceita	IHM aciona (1,0), recheio? (1), brigadeiro (1,0), cobertura? (1), sim(1,0), Confirmar pedido(1), pix(1), Confirmar Pagamento? (1),... bolo pronto(1,0), desligado
String Rejeitada	novo cliente (1,0), Recheio? (1), brigadeiro (1,0), cobertura? (1), sim(1,0), Confirmar pedido(1), pix(1), Confirmar Pagamento? (0)

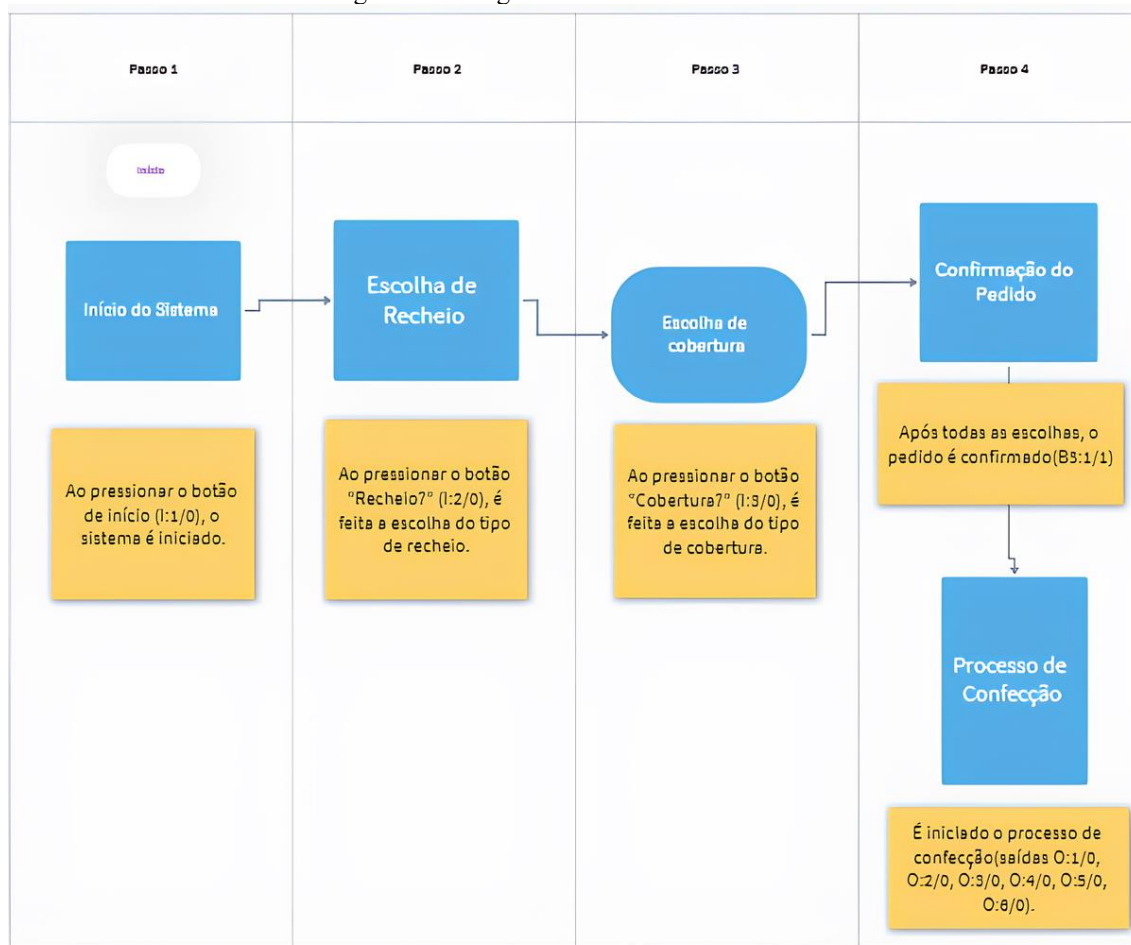
Essa sequência passa pelos estados relevantes de forma válida, resultando no estado final "Desligado/Reiniciado". Para isso ser válido, é necessário por restrição desse elemento, portanto o sétimo e o décimo primeiro elemento da string, deve ser 1, para poder ser feito o bolo. Caso o sétimo elemento seja 0, a string pode ter 8 elementos, para cada vez que o elemento 11 for 0, a cadeia deve ter dois elementos a mais na string. Além disso, a string deve ter no mínimo 21 elementos para terminar o processo. A String rejeitada é inválida, apesar de a sequência seguir o fluxo correto do processo, entretanto, não tem o número de elementos suficientes de strings para finalizar o processo. O sistema não pode ser desligado por um novo cliente sem passar pelos estágios apropriados.

#### 4.9 ESTRUTURAS DA LINGUAGEM LADDER PARA O SISTEMA DA MÁQUINA DE BOLOS

A linguagem de programação Ladder (LADDER Logic) desempenha um papel crucial nos controladores lógicos programáveis (CLPs) utilizados para automação industrial. No entanto, ao tentar traduzir modelos de Redes de Petri ou Autômatos diretamente para Ladder, depara-se com a complexidade decorrente das abordagens conceituais distintas entre essas metodologias. Neste contexto, buscamos desenvolver uma representação simplificada em Ladder que encapsula alguns elementos-chave provenientes da modelagem em Petri.

A tradução direta entre esses modelos pode encontrar limitações, dado que a Ladder é orientada à lógica de controle, enquanto as Redes de Petri são mais apropriadas para a modelagem de processos. A correlação entre ambos será, portanto, conceitual, e a implementação real dependerá da lógica específica de controle e das instruções disponíveis no ambiente Ladder utilizado. Lógica de Controle Simplificada em Ladder para o sistema proposto:

Figura 6. Fluxograma da estrutura do sistema.



Fonte: Imagem do autor

Os botões de escolha de recheio e cobertura são exemplos de entradas manuais, podendo ser sensores, botões ou entradas de outros dispositivos no ambiente real. A lógica aqui apresentada é simplificada e não contempla todas as condições ou detalhes específicos do sistema.

## 5 PROPOSTAS DE MELHORIAS PARA O SISTEMA

Uma proposta para aprimorar significativamente o sistema em questão envolve uma análise detalhada e expansiva do gerenciamento de pedidos, com foco especial no processo de pagamento. As modificações sugeridas abaixo buscam tornar a modelagem mais abrangente e eficiente, proporcionando uma visão mais clara e detalhada do fluxo de trabalho:

1. Inclusão de um Subprocesso de Pagamento Detalhado: aprimorar a representação do processo de pagamento através da adição de lugares e transições específicos. Isso incluiria a diferenciação entre opções de pagamento, como cartão de crédito, débito ou pix. Essa abordagem permitirá uma compreensão mais precisa e granular do fluxo de pagamento.

2. Introdução de Múltiplos Estados de Pagamento: expandir a modelagem ao criar lugares adicionais que representem diferentes estados do processo de pagamento. Exemplos incluem "Aguardando Confirmação do Cartão" ou "Aguardando Pagamento por Pix". Essa inclusão proporcionará uma visão mais abrangente do status de cada transação, facilitando a identificação de possíveis pontos de melhoria.
3. Adição de Feedback Visual no Diagrama: aprimorar a clareza do diagrama incluindo feedback visual para indicar estados específicos do sistema. A sugestão é utilizar cores para destacar transições bem-sucedidas, estados de espera ou estados finais. Essa abordagem proporcionará uma compreensão rápida e intuitiva do status do sistema.
4. Identificação de Possíveis Gargalos: realizar uma análise minuciosa da modelagem em busca de possíveis gargalos no processo. Além disso, fornece informações sobre tempos de execução ou outros parâmetros relevantes que possam indicar áreas específicas para aprimoramento. Essa abordagem visa otimizar a eficiência do sistema.
5. Consideração de Concorrência: se o sistema envolver a manipulação simultânea de vários pedidos, recomenda-se a inclusão de elementos que representem a concorrência. Isso poderia incluir, por exemplo, a confecção de vários bolos ao mesmo tempo. Essa consideração é crucial para sistemas que lidam com múltiplas operações simultâneas. A implementação dessas modificações na modelagem da Rede de Petri dependerá dos detalhes específicos do sistema em questão e dos requisitos funcionais associados. Ressalta-se a importância de adaptar as sugestões de acordo com a complexidade e os objetivos do sistema que está sendo modelado, assegurando uma abordagem personalizada e eficaz.

## 6 CONSIDERAÇÕES FINAIS

Após a implementação do sistema de redes de Petri e autômatos finitos, observou-se uma significativa melhoria no desempenho operacional da padaria. Anteriormente, o tempo de produção total para um bolo de pote era de aproximadamente 8 minutos, levando em consideração o transporte e despejo do creme e do recheio, e da interação usuário máquina. Com as novas estratégias implementadas, o tempo de operação foi reduzido para 5 minutos e 40 segundos, resultado da melhoria de eficiência ao administrar os processos de adição de ingredientes da máquina e no processo de adquirir o pedido do cliente, resultando em uma eficiência notável, que impactou diretamente no consumo kw/h da máquina e no uso desnecessário de materiais.

## 6.1 CÁLCULO DA EFICIÊNCIA MELHORADA

Potência da Máquina = 1500W

Preço Kw/h = 0,80c

Antes

8min = 0,13h

$$0,13h \times 1500W = 195W/Ciclo$$

Ciclo de operação = 195W

$$195 \frac{W}{C} \times 80\text{centavos KW/h} = 156\text{centavos}$$

Depois

5min40s = 0,09h

$$0,13h \times 1500W = 135W/Ciclo$$

Ciclo de operação = 135W

$$135 \frac{W}{C} \times 80\text{centavos KW/h} = 108\text{centavos}$$

Essa otimização teve um impacto direto na produção diária, permitindo a fabricação de mais bolos em um período de tempo menor. Antes, o processo aproveitava um período de 16h ao dia, totalizando aproximadamente 4 horas de produção retida. Agora, uma maior quantidade de bolos pode ser produzida em um tempo reduzido de 3 horas, liberando 4 horas adicionais de tempo de produção, reduzindo o custo de operação gerado pelo tempo parado. Além disso, a interação do cliente com a máquina foi aprimorada, proporcionando uma experiência mais fluida e eficiente. Esses resultados destacam a eficácia da implementação do sistema de redes de Petri e autômatos finitos na otimização dos processos de produção em uma máquina de confecção e venda de bolos, evidenciando ganhos significativos em termos de tempo e eficiência operacional.

Antes da aplicação do sistema:

$$30 \text{ dias} \times \frac{16h}{\text{dia}} = 480 \text{ por mês}$$

Depois da aplicação do sistema:

$$30 \text{ dias} \times 20h/\text{dia} = 600h/\text{mês}$$

### **AGRADECIMENTO**

O presente artigo é decorrente do projeto de Pesquisa e Desenvolvimento (P&D) ARANOUÁ, que conta com financiamento da Samsung, usando recursos da Lei de Informática para a Amazônia Ocidental (Lei Federal nº 8.387/1991), estando sua divulgação de acordo com o previsto no artigo 39.º do Decreto nº 10.521/2020.



## REFERÊNCIAS

HOPCROFT, John E.; MOTWANI, Rajeev; ULLMAN, Jeffrey D. Automata theory, languages, and computation. International Edition. Boston: Addison-Wesley, 2006. ISBN 978-0321455369.

SENA, Luciana de A.; MIYAGAWA, Makoto; SILVA JÚNIOR, Waldir S. da. Sistema de venda automática utilizando pagamento eletrônico através de dispositivos móveis. In: CONGRESSO NACIONAL DE INICIAÇÃO CIENTÍFICA - CONIC-SEMESP, 1., 2013, São Paulo. Anais. São Paulo: CONIC-Semesp, 2013. v. 1. Disponível em: <https://conic-semesp.org.br/anais/files/2013/trabalho-1000015410.pdf>. Acesso 01 julho 2024.

FRANCO, Thiago Almeida. Projeto para manutenibilidade de máquinas automáticas. 2007. 182 f. Tese (Doutorado em Engenharia Mecânica) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2007. Disponível em: <https://www.teses.usp.br/teses/disponiveis/3/3151/tde-09032023-084553/publico/TiagoAlmeidaFrancoOrig.pdf>. Acesso em: 07 julho 2024.

BARROS, Tomaz de Carvalho et al. Supervisão de sistemas de produção baseada em redes de Petri. Relatório técnico. [S.l.], 2000. Disponível em: <https://dspace.sti.ufcg.edu.br:8080/jspui/handle/riufcg/3461>. Acesso em: 15 julho 2024.

SCHEFFLER, Jordan Bönnmann; SILVA, Manoel Lenz Vianna da. Sistema de automação industrial para dosagem e mistura para sólidos granulados. [S.l.: s.n.], 2021. Disponível em: <https://www.publicacoeseventos.unijui.edu.br/index.php/salaoconhecimento/article/view/20555>. Acesso em: 17 julho 2024.