

UNIVERSIDADE CANDIDO MENDES – UCAM  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO  
CURSO DE MESTRADO EM ENGENHARIA DE PRODUÇÃO

Gustavo Rodrigues Fraga

SIMULAÇÃO COM OTIMIZAÇÃO EM TEMPO REAL EM PROCESSO  
DE DOSAGEM UTILIZANDO UM SOFTWARE LIVRE

CAMPOS DOS GOYTACAZES, RJ.  
Maio de 2017

UNIVERSIDADE CANDIDO MENDES – UCAM  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO  
CURSO DE MESTRADO EM ENGENHARIA DE PRODUÇÃO

Gustavo Rodrigues Fraga

SIMULAÇÃO COM OTIMIZAÇÃO EM TEMPO REAL EM PROCESSO  
DE DOSAGEM UTILIZANDO UM SOFTWARE LIVRE

Dissertação apresentada ao Programa de Pós-graduação em Engenharia de Produção, da Universidade Candido Mendes – Campos / RJ, para obtenção do grau de MESTRE EM ENGENHARIA DE PRODUÇÃO.

Orientador: Prof. João José de Assis Rangel, DSc.

Coorientador: Prof. Tulio Almeida Peixoto, MSc.

CAMPOS DOS GOYTACAZES, RJ.  
Maio de 2017

## FICHA CATALOGRÁFICA

F811s Fraga, Gustavo Rodrigues.

Simulação com otimização em tempo real em processo de dosagem utilizando um software livre. /. Gustavo Rodrigues Fraga – 2017.

f. il.92

Orientador: João José de Assis Rangel

Coorientador: Túlio Almeida Peixoto

Dissertação apresentado ao Curso de Mestrado em Engenharia de Produção da Universidade Candido Mendes - Campos dos Goytacazes, RJ, 2017.

Bibliografia: f. 79-84

1: Otimização (informática). 2. Simulação. 3. Controlador Lógico Programável (CLP). I. Universidade Candido Mendes – Campos. II. Título.

CDU – 004.383.4: 004.451.9

GUSTAVO RODRIGUES FRAGA

SIMULAÇÃO COM OTIMIZAÇÃO EM TEMPO REAL EM PROCESSO  
DE DOSAGEM UTILIZANDO UM SOFTWARE LIVRE

Dissertação apresentada ao Programa de Pós-graduação em Engenharia de Produção, da Universidade Candido Mendes – Campos / RJ, para obtenção do grau de MESTRE EM ENGENHARIA DE PRODUÇÃO.

Aprovado em maio de 2017.

BANCA EXAMINADORA

---

Prof. João José de Assis Rangel, DSc - Orientador  
Universidade Candido Mendes

---

Prof.: Túlio Almeida Peixoto, MSc. - Coorientador  
Universidade Candido Mendes

---

Prof.: Ítalo de Oliveira Matias, DSc.  
Universidade Candido Mendes

---

Prof. Fabiano Leal, DSc.  
Universidade Federal de Itajubá

CAMPOS DOS GOYTACAZES, RJ.  
2017

## **AGRADECIMENTOS**

Agradeço, primeiramente, a Deus por ter me abençoado e tornado possível esse sonho.

À minha família pelo amor, confiança e suporte prestado.

Ao meu orientador, João Rangel, e coorientador, Tulio Peixoto, por toda a paciência e atenção que tiveram comigo.

A todos os amigos que estiveram comigo durante toda esta caminhada.

## RESUMO

### SIMULAÇÃO COM OTIMIZAÇÃO EM TEMPO REAL EM PROCESSO DE DOSAGEM UTILIZANDO UM SOFTWARE LIVRE

Este trabalho tem como objetivo implementar um sistema de controle *online* que tenha capacidade de realizar ajustes na produção em tempo real usando a simulação a eventos discretos combinado com otimização algorítmica e transferência de dados com Controlador Lógico Programável (CLP). Foi desenvolvido um modelo hipotético de uma usina asfáltica, a programação de um CLP virtual para controlar o processo e a programação interna em linguagem JAVA do *software* de simulação que possibilita o envio de informações para Excel e execução de macros, com o objetivo de melhorar o desempenho do processo. A melhoria da performance nos processos industriais tem sido foco desde as primeiras revoluções industriais e com a chegada da era Industrial 4.0 ganhou ainda mais força. O termo “Indústria 4.0” compreende diversas tecnologias que permitem o desenvolvimento da fabricação automatizada e digital, sempre buscando a maior produtividade. Com o aumento do poder computacional, as ferramentas de simulação vêm ganhando espaço auxiliando na análise avançada e melhoria dos processos. Porém, um dos problemas para empresas de pequeno e médio porte é o custo elevado das ferramentas e técnicas de simulação. Tendo em vista esta dificuldade, este trabalho utilizou um software gratuito, livre e de código aberto para simular o processo industrial, buscando mostrar mecanismos empregados. O processo analisado na usina de asfalto hipotética foi a de dosagem de insumos frios. A ferramenta Solver do Excel integrada ao software de simulação a eventos discretos foi utilizada para encontrar a dosagem ótima de insumos em tempo real. Quando o processo é detectado fora de controle inicia-se a otimização que faz os cálculos da porcentagem ideal de cada produto, de acordo com a granulação dos insumos medida naquele momento. Foram realizados testes em diferentes cenários, variando o comportamento dos insumos do processo e variando os limites de controle. Os resultados obtidos demonstraram que o controle proposto pode ser efetivo e resultar em uma melhoria na qualidade do produto, mantendo-o dentro das especificações na maior parte do tempo. Além disso, a solução se mostrou simples e acessível por utilizar o Excel e um software de simulação livre e de código aberto, sendo esta a principal contribuição da pesquisa.

**PALAVRAS-CHAVE:** Otimização em tempo real. Simulação. Controlador Lógico Programável (CLP). Excel. Indústria 4.0.

## **ABSTRACT**

### **SIMULATION OPTIMIZATION IN DOSING PROCESS CONTROL SYSTEM IN REAL TIME IN A FREE OPEN SOURCE SOFTWARE**

This work aims to implement an online control system that has the ability to make real-time production adjustments using discrete event simulation combined with algorithmic optimization and data transfer with Programmable Logic Controller. A hypothetical model of an asphalt plant was developed, the programming of a virtual PLC to control the process and the internal programming in JAVA language of the software of simulation that allows the sending of information to Excel and execution of macros, with the objective of improving the process performance. The performance improvement in industrial processes has been focus since the first industrial revolutions and with the arrival of the Industrial era 4.0 gained even more force. The term Industrial 4.0 includes several technologies that allow the development of automated and digital manufacturing, always seeking the highest productivity. With the increase of computational power, the simulation tools have been gaining space, aiding in the advanced analysis and improvement of the processes. But one of the problems for small and medium-sized companies is the high cost of simulation tools and techniques. In view of this difficulty, this work used Free Open Source Software to simulate the industrial process, seeking to show mechanisms employed. The process analyzed in this work was the dosage of aggregates in plant of asphalt production. The Excel Solver tool integrated with the Discrete Event Simulation Software was used to find the optimal dosing of aggregates in real time. When the process is detected out of control it starts the optimization that makes the calculations of the ideal percentage of each aggregate, according to the granulation of the inputs measured at that moment. Tests were performed in different scenarios, varying the behavior of the process inputs and varying the control limits. The results showed that the proposed control can be effective and result in an improvement in product quality, keeping it within specifications most of the time. In addition, the solution was simple and accessible by using Excel and Free Open Source Simulation Software, which is the main contribution of this work.

**KEYWORDS:** Real time optimization; Simulation; PLC; Excel; Industrial 4.0.

## LISTA DE FIGURAS

<b>Figura 1.</b>	Exemplo de modelo construído no software Arena	31
<b>Figura 2.</b>	Arquitetura do Ururau.	41
<b>Figura 3.</b>	Ambiente de desenvolvimento do Ururau.	43
<b>Figura 4.</b>	Interface do Software RSEmulate 5000.	45
<b>Figura 5.</b>	Programação em Ladder com RSLogix 5000.	46
<b>Figura 6.</b>	Comunicação OPC do Ururau.	47
<b>Figura 7.</b>	Exemplo de aplicação da ferramenta Solver do Excel.	48
<b>Figura 8.</b>	Ativação do Solver no Excel.	49
<b>Figura 9.</b>	Ativação do Solver no VBA.	50
<b>Figura 10.</b>	Programação VBA que aciona o Solver.	50
<b>Figura 11.</b>	Ilustração do ambiente de desenvolvimento do software Factory Talk View Studio.	51
<b>Figura 12.</b>	Representação de uma mistura asfáltica típica.	52
<b>Figura 13.</b>	Típica Usina de Asfalto.	53
<b>Figura 14.</b>	Fluxo típico de análises do processo de Usina Asfáltica.	54
<b>Figura 15.</b>	Configuração do sistema de análise de imagem.	55
<b>Figura 16.</b>	Configuração do sistema de controle por análise de imagem.	57
<b>Figura 17.</b>	Integração da modelagem de simulação com o Excel Solver.	58
<b>Figura 18.</b>	Integração do modelo de simulação, Ururau, Excel e CLP.	61
<b>Figura 19.</b>	Ler/Escrever e Executar Macro na Interface do Ururau.	62
<b>Figura 20.</b>	Ilustração da Interface Homem-Máquina da Usina de Asfalto	67
<b>Figura 21.</b>	Modelo conceitual em IDEF-SIM.	68
<b>Figura 22.</b>	Captura de tela dos gráficos da simulação para Baixa Variação e	71

com a solução do Solver ativa.

**Figura 23.** Parte do relatório gerado pelo Ururau.

73

## LISTA DE FLUXOGRAMA E GRÁFICOS

<b>Fluxograma 1.</b>	Classificação dos sistemas para modelagem e simulação	27
<b>Gráfico 1.</b>	Número de publicações por ano	21
<b>Gráfico 2.</b>	Número de publicações por países.	21
<b>Gráfico 3.</b>	Diagrama de Componentes do Ururau.	22
<b>Gráfico 4.</b>	Representação Gráfica dos Sistemas Discretos e Contínuos: (a) Exemplo de Modelo de Sistemas Discreto; (b) Exemplo de Modelo de Sistema Contínuo.	27
<b>Gráfico 5.</b>	Função de Passo Alternado.	69

## LISTA DE QUADROS

<b>Quadro 1.</b>	Chave de Busca obtida para Seleção dos Artigos	20
<b>Quadro 2.</b>	Diagrama dos Componentes do Ururau	42
<b>Quadro 3.</b>	Relatório com os Resultados da Simulação	44
<b>Quadro 4.</b>	Produções Originadas neste Trabalho.	78

## LISTA DE TABELAS

<b>Tabela 1.</b>	Resultados obtidos por termo de pesquisa na base Scopus	20
<b>Tabela 2.</b>	Dados de entrada e parâmetros dos agregados	58
<b>Tabela 3.</b>	Dados iniciais e limites de abertura dos silos.	60
<b>Tabela 4.</b>	Dados de Baixa Variação e Alta Variação nos insumos.	64
<b>Tabela 5.</b>	Dados dos limites de controle utilizados.	65
<b>Tabela 6.</b>	Combinação dos Cenários.	65
<b>Tabela 7.</b>	Resultados das simulações.	74

## LISTA DE EQUAÇÕES

<b>Equação 1.</b>	Definição do Algoritmo de Otimização (1)	70
<b>Equação 2.</b>	Definição do Algoritmo de Otimização (2).	70
<b>Equação 3.</b>	Definição do Algoritmo de Otimização (3).	70
<b>Equação 4.</b>	Definição do Algoritmo de Otimização (4).	70
<b>Equação 5.</b>	Definição do Algoritmo de Otimização (5)	71

## LISTA DE ABREVIATURAS E SIGLAS

API.	Application Programming Interface
CLP.	Controlador Lógico Programável
COTS.	Commercial-Off-The-Shelf
FOSS.	Free and Open-Source Software
GPSS.	General Purpose Simulation System.
GSP.	General Simulation Program.
IHM.	Interface Homem-Máquina.
JSL.	Java Simulation Library.
OPC.	Ole for Process Control.
PLC.	Programmable Logic Controller
PM.	Projeto de Mistura
SCADA.	Sistemas de Supervisão e Aquisição de Dados (Supervisory Control and Data Acquisition)
SED.	Simulação a Eventos Discretos
VBA.	Visual Basic for Applications
VBS.	Visual Basic Script
WSC.	Winter Simulation Conference

## SUMÁRIO

<b>1.</b>	<b>INTRODUÇÃO</b>	17
1.1.	CONTEXTUALIZAÇÃO	17
1.2.	OBJETIVOS	18
<b>1.2.1.</b>	<b>Objetivo Geral</b>	18
<b>1.2.2.</b>	<b>Objetivos Específicos</b>	18
1.3.	JUSTIFICATIVAS	19
1.4.	QUESTÕES DE PESQUISA	23
1.5.	DELIMITAÇÕES DA PESQUISA	23
1.6.	ESTRUTURA DO TRABALHO	24
<b>2.</b>	<b>REVISÃO DA LITERATURA</b>	25
2.1.	SIMULAÇÃO COMPUTACIONAL	25
<b>2.1.1.</b>	<b>Vantagens e Desvantagens da Simulação</b>	28
2.1.1. 1.	Vantagens	28
2.1.1. 2.	Desvantagens	29
<b>2.1.2.</b>	<b>Softwares de Simulação</b>	30
2.2.	ESTADO DA ARTE	32
2.3.	CONCLUSÃO DA REVISÃO DA LITERATURA	38
<b>3.</b>	<b>MATERIAIS E MÉTODOS</b>	39
3.1.	METODOLOGIA DE SIMULAÇÃO E OTIMIZAÇÃO PROPOSTA	39
3.2.	EQUIPAMENTO E SOFTWARE DE SIMULAÇÃO	40
<b>3.2.1.</b>	<b>Software Ururau</b>	41
<b>3.2.2.</b>	<b>Equipamento de Controle</b>	44
<b>3.2.3.</b>	<b>Comunicação do Ururau com Controlador Lógico</b>	46

	<b>Programável</b>	
<b>3.2.4.</b>	<b>Ferramenta Solver do Excel</b>	47
3.2.4.1.	Ativar o Suplemento Solver no Excel	48
3.2.4.2.	Ativar a Referência de Programação do Solver VBA do Excel	49
3.2.4.3.	Inserir Código VBA d=que adiciona o Solver	50
<b>3.2.5.</b>	<b>Sistema de supervisão e aquisição de dados</b>	51
3.3.	DESCRIÇÃO DO SISTEMA	52
<b>3.3.1.</b>	<b>Processo de uma Usina de Asfalto</b>	52
<b>3.3.2.</b>	<b>Sistema de Controle em Tempo Real</b>	56
<b>3.3.3.</b>	<b>Dados de Entrada e Parâmetros</b>	59
<b>3.3.4.</b>	<b>Integração do modelo de simulação com o CLP e o Excel</b>	61
3.4.	DESCRIÇÃO DOS CENÁRIOS	63
<b>4.</b>	<b>RESULTADOS E DISCUSSÃO</b>	66
4.1.	MODELO DE SIMULAÇÃO	66
4.2.	MODELO DE OTIMIZAÇÃO	70
<b>4. 2.1</b>	<b>Porcentagem do Peso da Mistura Total vindo do Silos (xi)</b>	71
4.3.	EXPERIMENTOS SIMULADOS	71
<b>5.</b>	<b>CONSIDERAÇÕES FINAIS SOBRE A DISSERTAÇÃO</b>	76
5.1.	CONCLUSÕES	76
5.2.	LIMITAÇÕES DO TRABALHO	77
5.3.	SUGESTÃO PARA TRABALHOS FUTUROS	77
5.4.	PRODUÇÕES ORIGINADAS NESTE TRABALHO	78
<b>6.</b>	<b>REFERÊNCIAS BIBLIOGRÁFICAS</b>	79
	<b>APÊNDICE A: PARÂMETROS DA SIMULAÇÃO</b>	85
	<b>APÊNDICE B: PROGRAMAÇÃO EM LADDER</b>	87
	<b>APÊNDICE C: OTIMIZAÇÃO COM SOLVER DO EXCEL</b>	91

## 1. INTRODUÇÃO

### 1.1.CONTEXTUALIZAÇÃO

Xu *et al.* (2016), em um trabalho sobre simulação com otimização na era da Industrial 4.0, mencionou que uma nova era da industrialização tem surgido. O aumento da produtividade tem sido foco de todas as revoluções industriais anteriores, começando com a invenção da máquina a vapor na primeira revolução industrial e continuou com o Taylorismo, automação e informatização, sendo impulsionadas pelas próprias indústrias de produção. De acordo com Oesterreich (2016) e Schuh (2015), o termo Indústria 4.0 compreende as diversas tecnologias que permitem o desenvolvimento da fabricação automatizada e digital com intensa comunicação entre os equipamentos nos processos industriais.

Neste contexto, a modelagem e simulação têm sido descritas como um dos conceitos mais relevantes para gerenciar o aumento da complexidade dos processos de fabricação e auxiliando a melhorá-los. Desde a sua criação, a simulação tem sido aplicada a diversos setores, como manufatura, serviços, saúde e serviços públicos (JAHANGIRIAN, 2010). A simulação é utilizada tradicionalmente como uma ferramenta de análise, para prever o efeito de modificações em sistemas existentes, ou auxiliando a previsão do desempenho nos projetos em fase de concepção.

Em busca da maior produtividade, estas alterações no sistema devem ser cada vez mais precisas, buscando a maior performance do sistema. Por isso, pode ser utilizada a simulação em conjunto com técnicas de otimização que buscam configurações aprimoradas do sistema com relação à medida de desempenho.

Geralmente são utilizados algoritmos que buscam aproximações consecutivas em busca da melhor solução. As técnicas mais usadas são: baseado em gradiente, algoritmos de busca aleatória, algoritmos evolucionários, aproximações baseadas em programação matemática e técnicas de busca estatística (FU *et al.* 2000). Estes algoritmos podem realizar modificações no processo em tempo real. Desta forma, a simulação com otimização em tempo real permite que a mesma se adapte continuamente às perturbações e variações na entrada, mantendo a qualidade e requisitos do produto final durante todo o período de produção.

## 1.2. OBJETIVOS

### 1.2.1. Objetivo Geral

Implementar e testar a integração entre um software livre de simulação a eventos discretos (SED), Controlador Lógico Programável (CLP) e otimização em tempo real. Esta otimização utiliza a ferramenta Solver do software Microsoft® Excel® e busca o ponto ótimo de ajuste do processo que será testado em conjunto com o modelo de simulação e comunicação de dados com o sistema de controle.

### 1.2.2. Objetivos Específicos

- (I). Desenvolver o mecanismo de troca de dados entre o software Ururau com o software Microsoft® Excel® e acionamento do Solver.
- (II). Realizar a modelagem de um sistema de dosagem em um software de simulação a eventos discretos livre (Ururau).
- (III). Desenvolver um banco de dados de endereços e programar o CLP para monitorar as variáveis do processo de Usina Asfáltica.
- (IV). Desenvolver uma Interface Homem-Máquina (IHM) em um software de supervisão e aquisição de dados (SCADA) para que toda a simulação seja acompanhada em tempo real pelo usuário.

(V). Testar a integração entre o software Ururau, o CLP, a IHM e aplicativo Solver do Excel otimizando o processo em tempo real.

(VI). Mostrar que o software livre de código aberto é capaz de interagir com sistemas de controle e que a otimização em tempo real executada pelo Solver é capaz de melhorar a performance de um processo de dosagem de agregados da Usina Asfáltica.

### 1.3. JUSTIFICATIVAS

A proposta de trabalho se justifica na importância de se conhecer os mecanismos utilizados para integrar um *software* de simulação com otimização e um sistema de controle. A utilização de um *software* livre (o Ururau) permite a exploração dos mecanismos mais internos que podem ser empregados nas novas tecnologias disponíveis na era da Indústria 4.0.

Ainda, buscando avaliar a amplitude do tema da pesquisa, foi realizado uma busca por trabalhos realizados na área. Como ferramenta utilizada para pesquisa e análise do desenvolvimento científico produzido na área abordada nesta dissertação foi utilizada a bibliometria. Esta é uma técnica de medição de índices de disseminação do conhecimento e de produção científica.

Os indicadores bibliométricos possibilitam a análise do desenvolvimento científico identificando diversos fatores como a análise cronológica das produções científicas, produtividade das instituições e dos autores, trabalhos mais relevantes, dentre outros. As informações coletadas podem nos proporcionar uma avaliação quanto as principais tendências e evolução durante o tempo (WOSZEZENKI, 2013).

A base de artigos pesquisada foi o Scopus<sup>1</sup>. Foram aplicados comandos booleanos para unir "discrete event simulation" e "PLC". Sendo assim, a chave de busca obtida para seleção dos artigos foi:

---

<sup>1</sup> Disponível em: < <https://www.scopus.com/>>. Acesso em: 20 mar 2016.

**"discrete event simul\*" AND "control system"**

**Quadro 1.** Chave de Obtida para Seleção dos Artigos  
Fonte: Base Scopus (2016).

Com esta chave de busca foram obtidos 440 artigos no total. Porém, foram realizadas diversas combinações de palavras chaves e obtido os resultados conforme Tabela 1.

**Tabela 1.** Resultados obtidos por termo de pesquisa na base Scopus

TERMOS DE PESQUISA	ARTIGOS ENCONTRADOS
"discrete event simul*"	5.448
"discrete event simul*" AND "control system"	440
"discrete event simul*" AND "PLC"	27
"simulation*" AND "control system"	75.905
"simulation*" AND "control system" AND "VBA"	10
"simulation*" AND "control system" AND "excel"	26

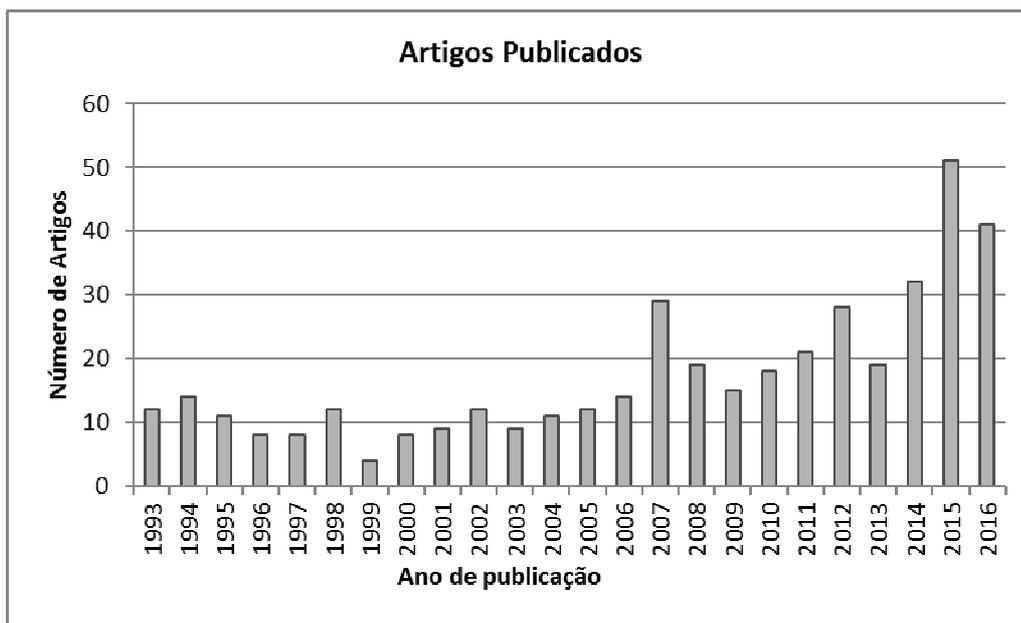
Fonte: Elaborado pelo Autor (2016).

Depois de delimitados os artigos que serão analisados, foram gerados vários gráficos, conforme mostrado a seguir.

O Gráfico 1 mostra o número de publicações do ano de 1993 até 2016. Pode-se observar o crescimento de publicações nesta área com o decorrer dos anos.

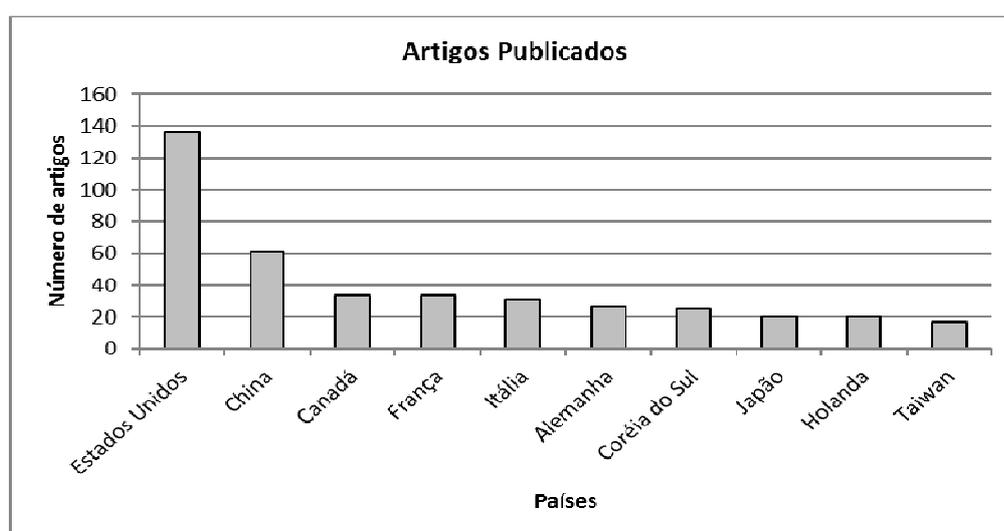
Os anos de 2015 e 2016 se destacam por ter mais publicações, 51 e 41 artigos publicados, respectivamente. O número de publicações somadas de 2010 até 2016 correspondem a mais da metade do total de artigos publicados.

A primeira publicação deste grupo de 440 artigos foi publicada no ano de 1975, revelando que nesta época já havia aplicabilidade da simulação a eventos discretos com sistemas de controle.



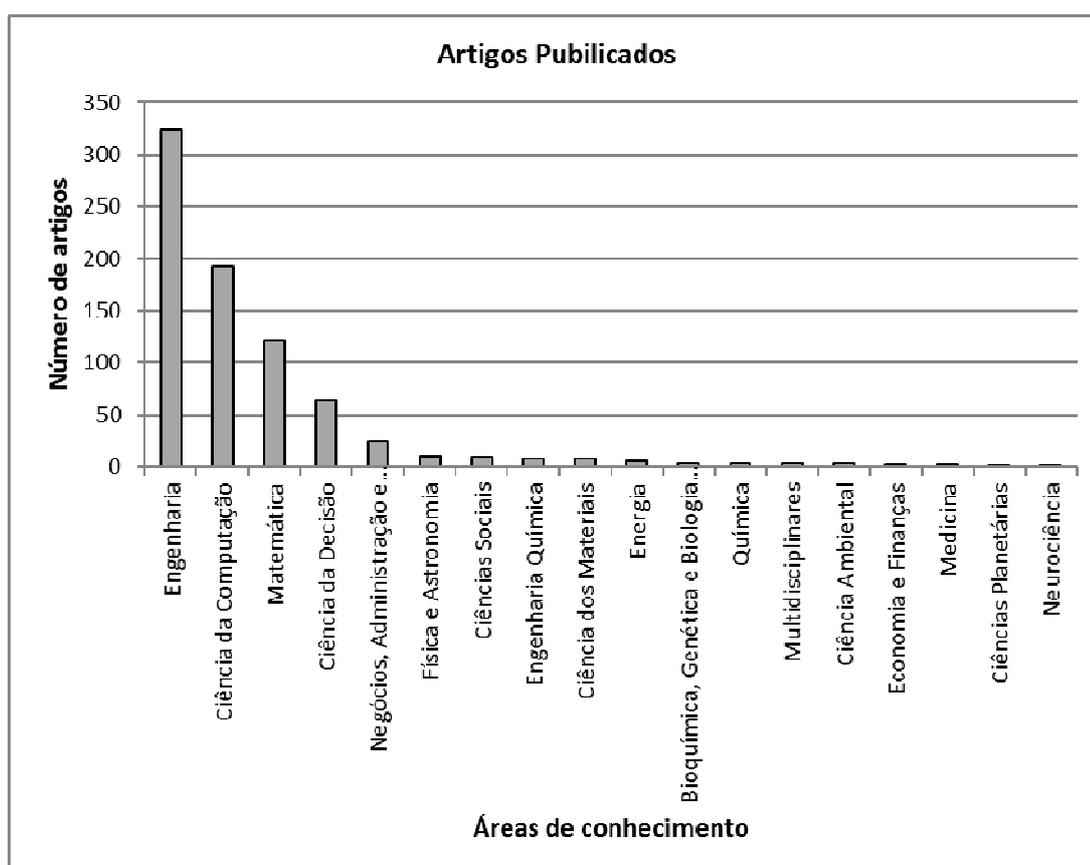
**Gráfico 1.** Gráfico do número de publicações por ano  
Fonte: Elaborado pelo Autor (2016).

Os países que mais contribuíram para as publicações foram os Estados Unidos e China. Os países supracitados publicaram 137 e 61 artigos, respectivamente. Pode-se observar que este tema ainda é pouco abordado no Brasil, pois nem entra na lista dos países que mais publicam sobre o tema. O Gráfico 2 mostra a distribuição das publicações por país de origem.



**Gráfico 2.** Gráfico do número de publicações por países  
Fonte: Elaborado pelo Autor (2016).

Os 440 artigos, então, foram agrupados em 18 áreas classificadas pela base Scopus, mostradas no Gráfico 3: Engenharia; Ciência da Computação; Matemática; Ciência da Decisão; Negócios, Administração e Contabilidade; Física e Astronomia; Ciências Sociais; Engenharia Química; Ciência dos Materiais; Energia; Bioquímica, Genética e Biologia Molecular; Química; Multidisciplinares; Ciência Ambiental; Economia e Finanças; Medicina; Ciências Planetárias; Neurociência



**Gráfico 3.** Gráfico do número de publicações por área de conhecimento  
Fonte: Elaborado pelo Autor (2016).

Observando o Gráfico 3, podemos notar que as áreas que se destacaram no tema de simulação a eventos discretos relacionados com sistema de controle é a de Engenharia e Ciência da Computação, com 324 e 192 publicações respectivamente.

#### 1.4. QUESTÕES DE PESQUISA

Diante do que foi exposto, as seguintes questões se apresentam como perguntas a serem respondidas neste trabalho, são elas:

- (A). O software de simulação “Ururau” tem a capacidade de simular processos industriais?
- (B). O software livre de código aberto Ururau tem a capacidade de trocar dados com planilha do Excel em tempo real durante a simulação?
- (C). É possível trocar dados eficientemente entre o software de simulação de eventos discretos “Ururau”, o CLP e o Microsoft Excel?
- (D). A otimização do Solver tem a capacidade de melhorar as performances dos processos de dosagem em tempo real?

#### 1.5. DELIMITAÇÕES DA PESQUISA

O trabalho de pesquisa desenvolvido nesta dissertação compreende os seguintes softwares e equipamentos:

- Modelo de simulação: Software Ururau;
- Modelo de otimização: Ferramenta Solver do Excel;
- Controlador: *Software* Emulador de CLP RSEmulate 5000;
- Interface Homem-Máquina: *Software FactoryTalk® View Studio*.

A pesquisa foi realizada integrando o software Ururau, com um CLP virtual via comunicação OPC, trocando dados com planilhas do Excel, executando macros que realizam cálculos de otimização e exibição dos dados em tempo real pelo software de supervisão e aquisição de dados. A pesquisa foi direcionada à aplicação desta solução no processo de dosagem de insumos frios de uma usina de asfalto, porém não foi testado o comportamento de toda a planta industrial.

## 1.6. ESTRUTURA DO TRABALHO

O trabalho desenvolvido foi estruturado da seguinte forma:

**(I).Capítulo 1 (Introdução):** apresenta a contextualização do trabalho, expondo pontos gerais da pesquisa, seus objetivos e justificativas. Foi realizado um levantamento bibliométrico para se verificar os principais trabalhos desenvolvidos sobre o tema. A seguir, foi exposto as delimitações da pesquisa e realizada uma descrição da estrutura do trabalho.

**(II). Capítulo 2 (Revisão da Literatura):** descreve pontos básicos da simulação, citando suas vantagens e desvantagens, *softwares* mais utilizados e aplicações mais comuns da ferramenta de simulação a eventos discretos. Também são descritos os principais artigos encontrados na pesquisa bibliométrica. Encerrando a seção, é feita uma conclusão com base nos artigos mais relevantes.

**(III). Capítulo 3 (Materiais e Métodos):** descreve todo o material utilizado na pesquisa e o método aplicado. Descrevendo o *software* de simulação a eventos discretos, o *software* de emulação de CLP, o *software* de comunicação, o sistema de supervisão e aquisição de dados, a ferramenta Solver do Excel e a interação entre eles. Também foi descrito o processo de usina de asfalto.

**(IV). Capítulo 4 (Resultados e Discussão):** foi apresentado o modelo conceitual e de simulação do processo de usina de asfalto, bem como os parâmetros e configurações utilizadas. Foi exposto os experimentos simulados e o resultado obtidos em cada cenário.

**(V). Capítulo 5 (Considerações Finais):** Este item são demonstradas as considerações finais do trabalho e responde as questões de pesquisa levantadas no início do trabalho. Também são apresentadas suas limitações, sugestões para trabalhos futuros e a produção originada.

## 2. REVISÃO DA LITERATURA

A seguir são apresentados os conceitos e fundamentos utilizados que servem de base para esta pesquisa. Foram descritos alguns conceitos básicos sobre simulação computacional, abordando as vantagens e desvantagens desta ferramenta. Posteriormente são apresentados alguns *softwares* de SED e algumas aplicações na área.

### 2.1. SIMULAÇÃO COMPUTACIONAL

A simulação computacional baseia-se na utilização de modelos matemáticos que empregados em computadores permitem imitar o funcionamento de diversos tipos de sistemas ou processos produtivos do mundo real (FREITAS FILHO, 2008).

Segundo Banks *et al.* (2010), a simulação é utilizada tradicionalmente como uma ferramenta de análise e design para prever o desempenho de sistemas ainda em concepção, ou para prever o efeito de mudanças em sistemas existentes sem necessariamente intervir no sistema real.

Ramadge e Wonham (1989), que foram pioneiros, sendo um dos primeiros a publicar artigos envolvendo o tema de simulação a eventos discretos integrado à sistema de controle. Eles descrevem SED como um sistema dinâmico que evolui de acordo com a ocorrência abrupta, em intervalos irregulares possivelmente desconhecidos, de eventos físicos. Por exemplo, um evento pode corresponder a chegada ou partida de um consumidor a uma fila de espera, a conclusão de uma tarefa ou a falha de uma máquina de um sistema de fabricação.

Segundo Goldsman *et al.*(2010), a história da simulação pode ser dividida em três partes:

- A Era Pré-computador (1777 – 1945);
- O Período Formativo (1945 – 1970); e
- O Período de Expansão (1970 – 1982).

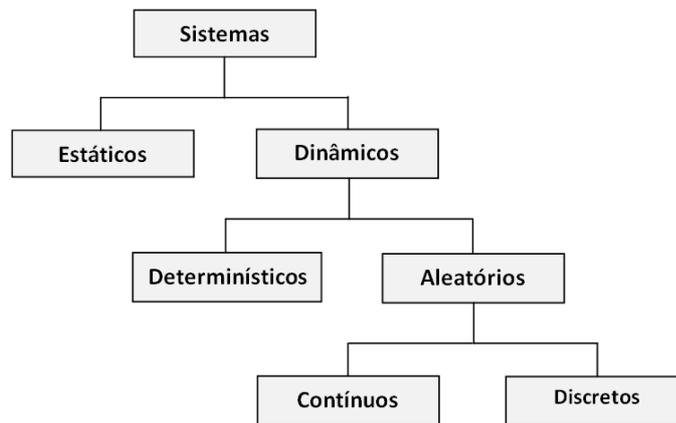
A Era Pré-Computador inicia-se com experimentos com simulações utilizando o método conhecido como Monte Carlo. Este método foi utilizado para calcular o valor de  $\pi$ . A simulação computacional em si começou durante a segunda guerra mundial, com os casos de modelos contínuos e de Monte Carlo, já a simulação discreta por volta de 1940.

O Período Formativo foi caracterizado pelo aumento da disponibilidade dos computadores eletrônicos, possibilitando a rápida difusão das técnicas de simulação. Nessa fase surgiu a primeira linguagem de simulação como GSP (*General Simulation Program*). A partir de então outras linguagens de simulação como GPSS (*General Purpose Simulation System*), o SIMSCRIPT e a SIMULA surgiram.

O Período de Expansão é marcado pelo avanço da ciência na simulação a eventos discretos, no que diz respeito ao ensino, pesquisa e prática. Nesta fase foi criado um dos mais importantes fóruns internacionais de simulação, o *Winter Simulation Conference* (WSC), divulgando os principais avanços nesta área.

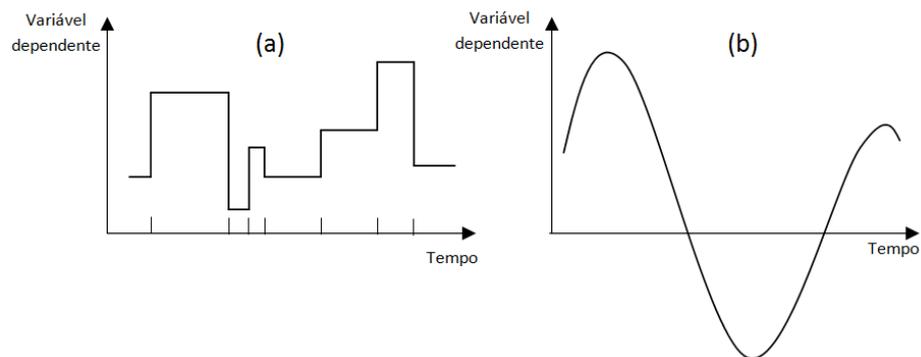
Sendo assim, a simulação a eventos discretos vem crescendo historicamente, sendo utilizada a fim de prever impactos de uma alteração no sistema, sem necessidade de modificação no sistema real.

Freitas Filho (2008) define sistemas como um grupo de componentes que recebem estímulos (entradas) e produzem respostas (saídas). De acordo com estes autores a classificação dos sistemas para fins de modelagem segue o esquema apresentado no Fluxograma 1.



**Fluxograma 1.** Classificação dos sistemas para modelagem e simulação.  
Fonte: Adaptado de Freitas Filho (2008)

Em um sistema estático as variáveis de estado não variam ao longo do tempo. Acontece o oposto com os sistemas dinâmicos, onde essas variáveis se alteram conforme a passagem do tempo. Os sistemas dinâmicos podem ser divididos entre determinísticos e aleatórios, onde nos sistemas determinísticos o comportamento é previsível e possível de se determinar, diferentemente dos sistemas aleatórios que apresentam comportamento de difícil previsão. Os sistemas aleatórios ainda podem ser subdivididos em mais duas partes: sistemas contínuos e sistemas discretos. Em sistemas contínuos as mudanças das variáveis de estado ocorrem de forma contínua no tempo, já nos sistemas discretos os eventos ocorrem em pontos específicos no tempo, conforme demonstrado graficamente pelo gráfico 4.



**Gráfico 4.** Representação gráfica dos sistemas discretos e contínuos: (a) exemplo de modelo de sistema discreto; (b) exemplo de modelo de sistema contínuo.

Fonte: Elaborado pelo Autor (2016).

Portanto, SED está classificado na análise de sistemas dinâmicos, aleatórios e discretos.

### **2.1.1. Vantagens e Desvantagens da Simulação**

A utilização da simulação gera diversos benefícios e vantagens. Porém, existem limitações que merecem ser destacadas. O autor Freitas Filho (2008), apresenta uma lista de alguns prós e contras do uso desta ferramenta, que são apresentados a seguir.

#### **2.1.1.1. Vantagens:**

(A). Após a criação do modelo computacional o mesmo pode ser utilizado inúmeras vezes para analisar projetos e políticas propostas;

(B). Geralmente, a simulação possui maior facilidade de aplicação em relação aos métodos analíticos;

(C). Os modelos analíticos necessitam de um número alto de simplificações para torná-los matematicamente viáveis, já os modelos de simulação não requerem um grande número de simplificações, desta forma é possível analisar praticamente qualquer medida concebível;

(D). Uma vez que os modelos representam o sistema real, as mudanças no sistema podem ser testadas sem que seja alterado o sistema real;

(E). Hipóteses sobre o porquê ou como certos fenômenos ocorrem podem ser testados para sua confirmação;

(F). É possível controlar a velocidade da reprodução dos fenômenos para melhor estudo do sistema;

(G). Podem-se compreender melhor as variáveis que têm maior impacto no sistema e suas relações;

- (H). Podem-se identificar mais facilmente os gargalos dos sistemas produtivos, principalmente com ajuda visual;
- (I). A simulação pode mostrar o contraste entre como o sistema opera de fato e como foi a idealização preliminar de seu funcionamento;
- (J). É possível prever como um novo sistema se comportará antes que este seja construído.

#### 2.1.1.2. Desvantagens:

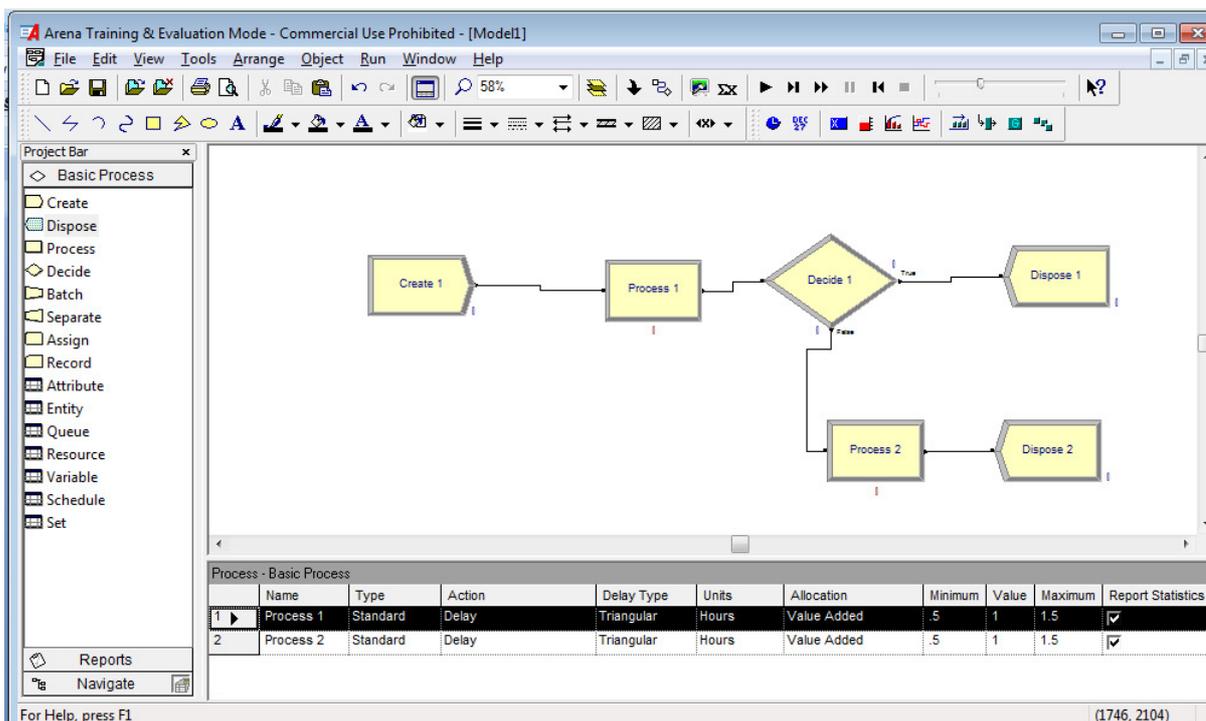
- (A). É necessário um treinamento especial para que o profissional esteja capacitado para construção dos modelos computacionais;
- (B). Possível dificuldade de interpretação dos resultados obtidos com a simulação;
- (C). Grande consumo de tempo e recursos na elaboração dos modelos e experimentos;
- (D). Não é possível aproveitar a modelagem de um sistema para se aplicar em outro diferente, a menos que seja continuação ou parte deste outro;
- (E). A simulação não gera bons resultados se o modelo não for elaborado corretamente, podendo gerar uma interpretação falsa do sistema;
- (F). Apesar do uso de *softwares*, a simulação em si não é uma técnica de otimização;
- (G). Às vezes, outras soluções mais simples poderiam resolver o problema em menos tempo do que a simulação;
- (H). De acordo com a complexidade do sistema, a modelagem do mesmo pode ser tornar grande e demasiadamente complexo;
- (I). Necessidade de equipamentos potentes de *hardware* e *software* para execução e validação do modelo.

### 2.1.2. Softwares de Simulação

Existem diversos softwares de simulação no mercado, podendo ser comerciais (*Commercial-Off-The-Shelf* – COTS), livres de código aberto (*Free and Open-Source Software* – FOSS), ou ainda se apresentar em ambas versões. Os softwares de simulação podem ser aplicados em diversas áreas, como no início nas áreas militares, em precisão meteorológica, pilotagem de veículos, dentre outros, sempre nos auxiliando a estudar e prever o comportamento dos sistemas.

Existe um número muito grande de softwares no mercado e não faz parte do escopo da pesquisa discutir todos eles. Portanto, será exposto de forma breve nesta seção o *software* comercial Arena. Este *software* é uma das ferramentas SED reconhecidas mundialmente e mais utilizadas no segmento.

O Arena é um *software* de simulação fornecido pela empresa *Rockwell Automation*, um fabricante tradicional e muito conhecido no segmento da automação. Foi um dos primeiros *softwares* a oferecer a funcionalidade de programação em alto nível e animação gráfica. O *software* permite a modelagem de sistemas de diversos segmentos, como processos de dosagem, logísticos e sistemas automatizados. É um dos pioneiros na integração da simulação com sistemas de controle devido a sua grande integrabilidade com os outros softwares e controladores da *Rockwell* (KELTON, 2010). Todo o processo de criação do modelo é feito através de blocos e caixas, facilitando o entendimento do fluxo de informações e processos nas empresas. O ambiente de simulação do Arena pode ser observado na Figura 1.



**Figura 1.** Exemplo de modelo construído no software Arena.  
Fonte: Elaborado pelo Autor (2016).

Além disso, o Arena inclui no seu pacote a ferramenta Input Analyser, que auxilia na determinação das curvas de comportamento dos sistemas. Esta ferramenta fornece uma expressão matemática que melhor descreve o comportamento dos dados.

O software possui um alto custo de licença e necessidade de renovação anual da mesma. Esse alto custo impossibilita, muitas vezes, a aquisição e utilização desta ferramenta por empresas de pequeno e médio porte. Portanto, os softwares livres representam uma alternativa para estes tipos de empresas, dando acessibilidade às ferramentas de simulação.

Segundo Dagkakis e Heavey (2015), existem 44 *softwares* de simulação livres no mercado. Os autores apresentaram uma revisão de literatura e listam 10 *softwares* livres de SED, descrevendo suas principais características, aplicabilidade, linguagem de programação utilizada, dentre outros. Os softwares listados foram: OMNet++C, NS-3C, SimPy, JaamSim, JAPROSIM, DESMO-J, Fascimile, SharpSim, PowerDEVS e Ururau.

O *software* Ururau, que foi listado no artigo supracitado, é o primeiro *software*

livre e de código aberto de SED desenvolvido no Brasil e foi utilizado nesta pesquisa para simular um processo de usina de asfalto (PEIXOTO *et al.* 2017). Maiores detalhes desta ferramenta estão descritas no item 3 (Materiais e Métodos).

## 2.2. ESTADO DA ARTE

A integração de simulação, com um sistema de controle e otimização é um assunto inovador e ainda pouco abordado. Este tema vem sendo cada vez mais explorado devido à crescente preocupação com os custos e em produzir cada vez mais com menos recurso e em menor tempo.

O levantamento bibliográfico realizado em fevereiro de 2017, na base de dados Scopus, retornaram alguns artigos que tiveram mais semelhança com a pesquisa desenvolvida. Os 3 artigos de base utilizados como referência da pesquisa são: PEIXOTO *et al.* (2017), ZHANG *et al.* (2014) e KABADURMUS *et al.* (2010). Estes artigos são brevemente descritos a seguir.

Peixoto *et al.* (2017) descrevem a arquitetura e a operação de um software de simulação a eventos discretos livre e de código aberto programado em Java. O software é conhecido como Ururau, desenvolvido por brasileiros. Os testes realizados neste produto tecnológico (SILVA *et al.* 2017) demonstraram que ele tem a capacidade de construir e simular modelos simples mas com grande agilidade e eficiência.

Em um trabalho de Kabadurmus *et al.* (2010) os autores descrevem um novo processo de controle em linha do sistema de produção de asfalto. Neste processo, um sistema de processamento de imagens analisa continuamente as amostras da entrada de insumo e da saída do produto processado, e as ações corretivas necessárias no controle são tomadas instantaneamente por um sistema de otimização computadorizada em VBA para encontrar a melhor mistura asfáltica. O modelo de simulação discreta do sistema permite testar o controle em diversos cenários. Essa simulação foi realizada no software Arena, onde se diferencia da pesquisa proposta que foi feita em um software totalmente livre e de código aberto. Os resultados demonstrados foram satisfatórios e o sistema se mostrou controlado

na maioria do tempo de simulação.

O artigo proposto por Zhang *et al.* (2014) é semelhante ao proposto por Kabadurmus *et al.* (2010), porém este artigo foi submetido a um periódico da área de engenharia civil (Journal of Computing in Civil Engineering) e nos dá detalhes como os parâmetros do processo da usina de asfalto que o outro trabalho não nos dá. Sendo assim, este trabalho possui dados essenciais que fundamentam a execução da pesquisa aqui realizada.

Estes três artigos nortearam a pesquisa, porém no levantamento bibliométrico foram encontrados diversos outros artigos, descritos a seguir, que auxiliaram a pesquisa e reforçam a importância do tema abordado.

Fabian e Hellgren (1998) descrevem a teoria de controle supervisório que é uma teoria geral para a síntese automática de controladores para sistemas a eventos discretos, dado um modelo de planta e uma especificação para o comportamento do controlador. O trabalho apresentado ressalta que esta teoria existe a bastante tempo, porém ainda existem poucas aplicações industriais naquela época. A principal razão para isto é a discrepância entre a simulação abstrata e sua implementação física. Este artigo chama atenção para os principais problemas de aplicação da simulação com o CLP e sugere procedimentos para atenuar o problema.

Alvarez e Centeno (1999) desenvolveram um modelo de simulação de uma sala de emergência de um hospital em conjunto com rotinas VBA para que ela possa ser utilizada no mundo real. O modelo de simulação foi complementado com uma série de rotinas de tomadas de decisão. Estas rotinas usam uma abordagem hierárquica para organizar os vários cenários em que o modelo pode funcionar e para reconfigurar parcialmente o modelo ARENA em tempo real. O modelo utilizado é facilmente configurável através de uma interface com o usuário, sem que o mesmo precise acessar o software de simulação.

O trabalho de Henriksen *et al.* (1999) desenvolve um sistema de orientação e treinamento de operador para um processo industrial que envolve ideias de programação de sistemas de manufatura. O objetivo do trabalho é aumentar o rendimento, melhorando a utilização dos recursos do processo. Este trabalho utiliza

a aplicação de técnicas de programação em conjunto com a simulação para otimizar a utilização de pontes rolantes da planta que movimentam material. O resultado foi capaz de gerar programações úteis e com um tempo de processamento computacional aceitável.

Seppanen (2000) elaborou um tutorial que demonstra os conceitos básicos desenvolvidos pelo autor para transferir dados entre o Arena® e o Excel®. As mesmas técnicas demonstradas podem ser utilizadas para diversas ferramentas suportadas pelo VBA (Visual Basic® for Applications), como Access®, AutoCAD® e Visio®. O foco do trabalho se concentra no desenvolvimento de planilhas do Excel para aplicações de simulação e a transferência de dados de/para o Arena usando VBA.

Schludermann (2000) aborda a utilidade da simulação para o comissionamento e verificação do comportamento do CLP. Foi desenvolvido uma arquitetura modular com um protocolo de comunicação comum a todos os elementos do sistema. Os módulos utilizados no sistema foram um driver de dispositivos de entradas e saídas chamada de IODD, que liga o I/O com o protocolo SoftCom, e a Interface de Simulação com o Mundo Real, chamado de SWI, que liga o simulador com o sistema SoftCom.

O artigo do Queiroz e Cury (2002) propõe uma integração entre a simulação de eventos discretos e o CLP. Aborda uma metodologia que explora a modularidade da planta proposta, seus modelos e especificações comportamentais através da simulação discreta. O comportamento da célula de manufatura flexível, após a aplicação do sistema de controle prático e a flexibilidade do código do CLP são pontos positivos para a metodologia aplicada. O novo sistema de controle implementado apresentou melhoras no comportamento da célula de manufatura e na leitura e flexibilidade do programa de CLP.

Outro trabalho de destaque na área é dos autores Jacobs e Verbraeck (2005) que utilizam um software de simulação desenvolvido em JAVA chamado DSOL (Distributed Simulation Object Library). O estudo de caso foi realizado em uma fábrica de piso de concreto. Os testes com o DSOL utilizaram protocolo de comunicação MODBUS com CLPs físicos e emulados por softwares. Diversos

aspectos inerentes à plataforma JAVA são tratados nesse trabalho, como bibliotecas e códigos utilizados.

Costa *et al.* (2005) apresentam uma proposta para implementação da teoria de controle supervisório modular local, focada no custo. Explora a realização do modelo teórico em plataformas de baixo custo, com base em CLPs de pequeno porte, micro-controladores e dispositivos eletrônicos comerciais. Uma estrutura de controle é, então, proposta como a solução adequada para pequenos e médios sistemas de manufatura automatizada. O modelo proposto proporcionou uma redução nos custos. O modelo proposto por Ramadge e Wonham (1989) e Queiroz e Cury (2000), são utilizados afim de fornecer um processo automatizado para a síntese do supervisório, em vez de os procedimentos manuais ou heurísticas habituais.

O trabalho de Wange e Takakuwa (2006) desenvolve uma modelagem baseada em módulo em conjunto com VBA do Excel que é utilizado para gerar automaticamente os programas de simulação. O modelo pode ser usado para analisar o desempenho de uma cadeia de suprimentos, como retornos dos custos de transporte e custo de inventário, níveis de serviço e controle de estoques. Aplicando a metodologia proposta foi possível construir rapidamente um sistema complexo e rapidamente alterar as configurações do sistema.

Johnstone *et al.* (2007) abordam a transferência de dados entre o CLP e o Computador via padrão OPC (OLE Process Control). Esta comunicação permite o teste de sistemas de controle com um modelo de emulação. A proposta do trabalho é o desenvolvimento de uma interface que permite a comunicação em alto volume entre o CLP e o PC, permitindo a emulação de sistemas maiores, mais complexos e com maior velocidade na transmissão de dados.

Foi proposto no trabalho de Park *et al.* (2008) uma arquitetura de um ambiente de programação de CLP que permite uma verificação visual dos programas de CLP. O modelo proposto integra um programa de CLP com um modelo de planta correspondente onde os usuários podem intuitivamente verificar o funcionamento do programa do CLP em um ambiente gráfico em 3D.

Smith e Cho (2008) abordam um método generalizado de mapear um modelo

de simulação para que seja utilizado um emulador de CLP e realizado testes de lógica e comissionamento off-line. Foi utilizado como exemplo um sistema de embalagem de alta velocidade integrado a aplicações da empresa Rockwell Automation. A simulação, controle e comissionamento do sistema proposto se mostraram bastante eficazes. Apesar de focar apenas nos softwares da Rockwell, o autor ressalta que esta metodologia pode ser aplicada a outras ferramentas de simulação e controle.

Mahdavi *et al.* (2010) também abordam este assunto sobre a utilização da simulação e otimização em um sistema de manufatura flexível. Este artigo descreve um Sistema de apoio à decisão baseada na simulação para controle de produção estocástico. O desenvolvimento do controle é baseado na simulação de eventos discretos em tempo real. O controlador recebe os resultados online do simulador e identifica oportunidade de melhoria incremental. Os autores desenvolveram uma simulação baseada em eventos discretos chamada de GEPSIS que integra uma linguagem de simulação em SIMPLE++ e EXCEL (VBA).

O trabalho de Rangel *et al.* (2012) apresenta a integração do CLP com simulação a eventos discretos de um sistema de transporte e seleção com esteira. O modelo foi desenvolvido no Arena e comparado com um modelo didático já existente chamado de ITS PLC que serve para treinamento e aprendizado da aplicação de lógica de programação de CLPs. Após os testes aplicados pôde-se constatar que o sistema desenvolvido se comportou semelhante ao sistema original de treinamento. O sistema proposto tem ainda a vantagem de poder ser alterada a configuração do sistema de acordo com o desejo do usuário, possibilitando uma diversidade de testes e cenários.

Cardoso *et al.* (2014) integram um software livre de simulação à eventos discretos “Ururau” com CLP. O sistema proposto simulou uma planta didática real e apresentou resposta semelhante à planta real. O trabalho traz uma inovação para o mercado nacional, pois introduz um software livre com a possibilidade de alteração, coisa que a maioria dos outros softwares do mercado não possibilita. O resultado apresentado neste estudo demonstrou a possibilidade de aplicar o Ururau como recurso didático de baixo custo.

Foi demonstrado no trabalho de Brooks *et al.* (2014) que o software de modelagem pode ser implementado em todas as fases do projeto, desde o layout e simulação até os testes do sistema de controle. Também mostra que a lógica de controle do sistema pode ser utilizada em todas as fases através da incorporação de um emulador de CLP, totalmente integrado com o relógio virtual e utilizando os canais de comunicação interna. O engenheiro tem total controle sobre o nível de precisão para a modelagem, simulação e emulação, além do controle da velocidade de execução do modelo.

Riera *et al.* (2015) apresentam um método não-invasivo para programar sistemas de segurança na manufatura para Controladores Lógicos Programáveis. Os sistemas de manufaturas são considerados como Sistemas de Eventos Discretos, com entradas lógicas (sensores) e saídas lógicas (atuadores). A ideia proposta pelo trabalho não é verificar um programa específico de CLP, mas adicionar no final do programa de CLP um algoritmo específico com base nas restrições de segurança que vai autorizar ou restringir as saídas do CLP, a fim de garantir a segurança. O algoritmo apresentado em detalhes é ilustrado com um sistema de 4 cilindros. Esta abordagem também pode ser utilizada para projetar programas de CLP de segurança, separando a parte funcional do processo da parte de segurança.

O artigo de Wilson *et al.* (2016) propõe ferramentas associadas a simulação de eventos discretos para auxiliar na gestão de energia em um sistema de produção. O kit de ferramentas utilizado tem base no software Microsoft Excel integrado com o software de simulação WITNESS utilizado para controlar parâmetros da simulação otimizando o sistema. Esta ferramenta desenvolvida tem capacidade de prever áreas potenciais onde pode ser feito a economia de energia dentro de uma linha de produção complexa de motores automotivos.

Os trabalhos supracitados reforçam a importância do tema e mostra que ainda é uma área que tem muito potencial a ser desenvolvido.

### 2.3. CONCLUSÃO DA REVISÃO DA LITERATURA

Com base nos artigos analisados conclui-se que a simulação de eventos discretos é de fundamental importância em diversos aspectos, como desenvolvimento e testes de sistemas, verificação de melhorias implementadas, dentre outros.

Na área da automação, a simulação integrada ao CLP tem demonstrado grande funcionalidade nos testes de lógica e comissionamento dos processos. Esta característica proporciona economia e segurança, pois antes mesmo do processo físico ser implantado, pode-se realizar diversos experimentos sem causar danos ou desperdícios nos sistemas reais.

A otimização vem sendo cada vez mais discutida com o objetivo de se produzir mais com menos recurso e no menor tempo. A simulação em conjunto com lógicas de otimização propicia arquitetar processos cada vez mais eficientes.

Neste contexto, é de grande relevância a integração da simulação, controle e otimização que caminha junto com filosofia enxuta, segurança e sustentabilidade.

### 3. MATERIAIS E MÉTODOS

#### 3.1. METODOLOGIA DE SIMULAÇÃO E OTIMIZAÇÃO PROPOSTA

A construção dos modelos de simulação seguiu a metodologia proposta por Banks *et al.* (2010), em que foram seguidos os seguintes passos: formulação do problema; definição do objetivo e plano geral do projeto; elaboração do modelo conceitual; coleta de dados; tradução do modelo conceitual; verificação; validação; experimentação; execução e análise; rodadas; documentação e relatório e implementação. A verificação e validação dos modelos seguiram as etapas sugeridas por Sargent (2013).

**(1). Formulação do problema:** o estudo foi iniciado com a declaração do problema a ser investigado, sendo descrito de maneira clara e entendível;

**(2). Definição do objetivo e plano geral do projeto:** definiram-se as questões a serem respondidas pela simulação, os custos relacionados, o tempo necessário para realizar cada fase do trabalho e os resultados esperados ao fim de cada etapa;

**(3). Elaboração do modelo conceitual:** as características essenciais dos sistemas foram elencadas para a construção dos modelos conceituais, neste caso, por meio da linguagem IDEF-SIM;

**(4). Coleta de dados:** os dados necessários para alimentar o modelo conceitual do sistema foram obtidos com artigos anteriores relacionados com a produção de asfalto.

- (5). **Tradução do modelo conceitual:** realizou-se a implementação dos modelos conceituais em um ambiente computacional, o *software* livre de simulação Ururau 1.1;
- (6). **Verificação:** a lógica dos modelos computacionais foi testada para verificar se o funcionamento do mesmo se mostrava coerente com o sistema investigado;
- (7). **Validação:** realizou-se a calibração dos modelos computacionais contrastando as respostas das simulações com o funcionamento dos sistemas;
- (8). **Experimentação:** definiram-se as alternativas simuladas e os parâmetros de execução (tempo e número de replicação);
- (9). **Execução e análise:** foi realizada uma rodada de simulação para cada situação de controle proposta para análise dos resultados;
- (10). **Rodadas:** verificou-se se os parâmetros da simulação foram suficientes e se havia necessidade de mais testes;
- (11). **Documentação e relatório:** foram elaborados os documentos e relatórios do modelo de simulação para posterior manipulação do analista ou outros e para viabilizar o desenvolvimento de modelo similar;
- (12). **Implementação:** com os modelos finalizados, após passar por todas as etapas já mencionadas, foi possível executá-los e obter os resultados para o problema declarado inicialmente.

A otimização proposta será iniciada quando for detectado que o sistema está fora de controle (fora dos limites de especificações definidos). A otimização consiste em executar o aplicativo Solver do Excel para que o mesmo encontre as variáveis ótimas do sistema para que o controle do processo seja retomado. O fluxograma de integração e funcionamento do sistema é demonstrado na Figura 17, na página 56

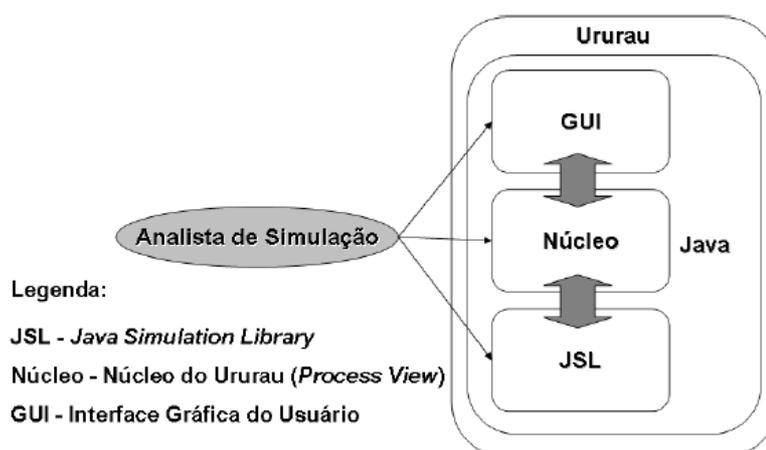
### 3.2. EQUIPAMENTO E SOFTWARE DE SIMULAÇÃO

A interface implementada integra o *software* Ururau, o CLP e a ferramenta Solver do Excel.

### 3.2.1. Software Ururau

O Ururau é um *software* desenvolvido no Brasil para elaboração e testes de modelos de simulação a eventos discretos. O sistema foi desenvolvido em linguagem Java orientado a objetos utilizando como base a biblioteca de simulação JSL – *Java Simulation Library*, proposto por Rossetti (2008). O ambiente permite flexibilidade, pois o *software* foi desenvolvido em código livre e aberto (*Free and Open Source Software* – FOSS) que permite alterações na programação e inclusões de novas rotinas.

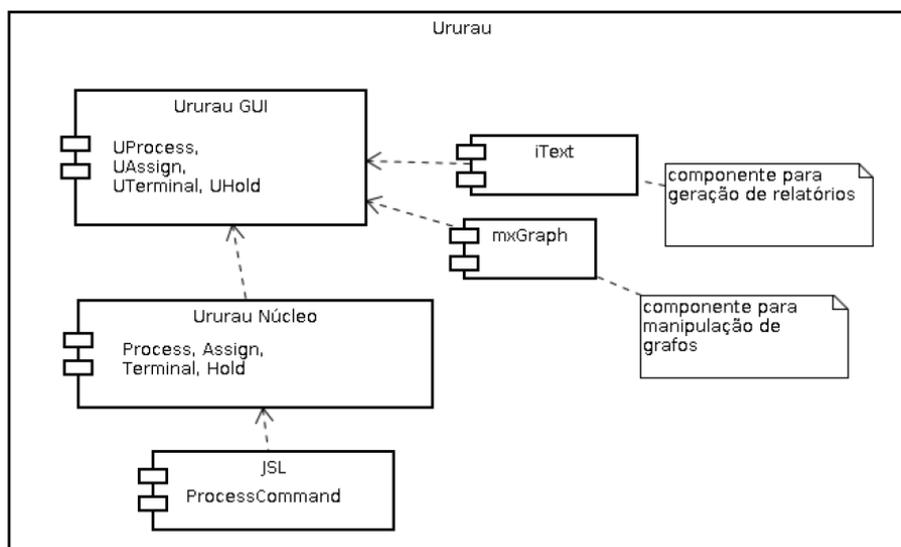
A Figura 2 apresenta a arquitetura do *software* Ururau. Em todas as camadas. A biblioteca JSL fica na camada mais inferior e converte o modelo, que é composto por uma série de comandos de processo (*Process View*), para uma sequência de eventos discretos. A camada intermediária é o núcleo do Ururau, que é composto por comandos de processos específicos do JSL, como criação de entidades, prender (*Seize*), atraso (*Delay*), liberar (*Release*), dentre outros. A camada mais superior trata da conversão do modelo gráfico, que é composto por um grafo dirigido para uma sequência de comandos do núcleo do Ururau.



**Figura 2.** Arquitetura do Ururau.  
Fonte: Peixoto (2017)

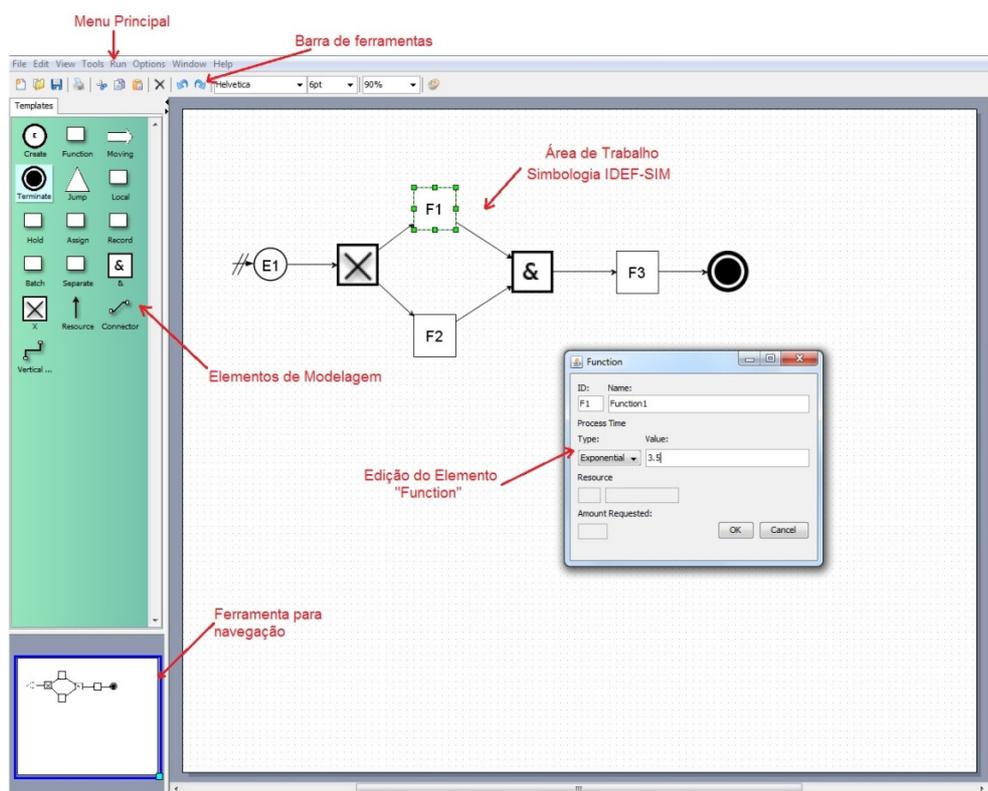
Os comandos do núcleo do Ururau se assemelham ao conjunto de *templates* básicos da maioria dos *softwares* de simulação. Além disso, o *software* tem uma interface gráfica que opera estes comandos que compõem o núcleo do *software*

que, por sua vez, opera o JSL, conforme mostra o quadro 2.



**Quadro 2.** Diagrama de componentes do Ururau.  
Fonte: Peixoto (2017)

A sua interface gráfica, conforme demonstra a Figura 3, permite o desenvolvimento de modelos de maneira simples e fácil através da simbologia baseada em IDEF-SIM, proposta por Montevechi *et al.* (2010).



**Figura 3.** Ambiente de desenvolvimento do Ururau.  
Fonte: Elaborado pelo Autor (2016).

Para a construção do modelo de simulação, o desenvolvedor precisa apenas arrastar e soltar com o mouse os elementos de modelagem para a área de trabalho, de acordo com a lógica do sistema em análise e simbologia IDEF-SIM. A seguir, faz-se a edição dos módulos clicando com o botão direito do mouse sobre o mesmo para abri-lo. Na caixa de edição, podem-se inserir dados do modelo como os tempos dos eventos, os tipos de distribuições, regras operacionais, entre outras funções.

Uma vez desenvolvido o modelo, são definidos os parâmetros da simulação. São eles: o número de replicações, o tempo de replicação e o tempo de aquecimento. A partir daí, o modelo é executado, e, ao final, um relatório com as estatísticas é gerado. Um modelo de relatório é apresentado no Quadro 3.

<b>Experiment Parameters</b>				
Experiment: Model-76				
Number of Replications: 5.0				
Length of Replication: 100.0				
Length of Warm Up: 0.0				
<b>User Variables</b>				
Other Variables	average	semi interval	minimum	maximum
<b>Entity</b>				
Flow Time	value			
Entity1	26.581			
<b>Fila</b>				
Time in Queue	average	semi interval	minimum	maximum
F2.Function2.queue	24.213	4.453	19.457	32.644
F3.Function3.queue	0.791	0.424	0.463	1.640
Number in Queue	average	semi interval	minimum	maximum
F2.Function2.queue	24.889	5.386	16.556	33.438
F3.Function3.queue	0.367	0.217	0.176	0.771
<b>Resource</b>				
Buzy	average	semi interval	minimum	maximum
R1 Resource1	0.988	0.011	0.969	1.000
R2 Resource2	0.497	0.078	0.372	0.621
Number Idle	average	semi interval	minimum	maximum
R1 Resource1	0.012	0.011	0.000	0.031
R2 Resource2	0.503	0.078	0.379	0.628

**Quadro 3.** Relatório com os resultados da simulação.  
Fonte: Elaborado pelo Autor (2016).

Para transferência de dados entre o Ururau e o Microsoft Excel foi utilizado a interface API (Application Programming Interface) do Java chamada de Apache POI. O objetivo desta interface é manipular diversos arquivos baseados no Open Office XML e OLE da Microsoft.

A documentação completa e o download das bibliotecas do Apache POI<sup>2</sup>

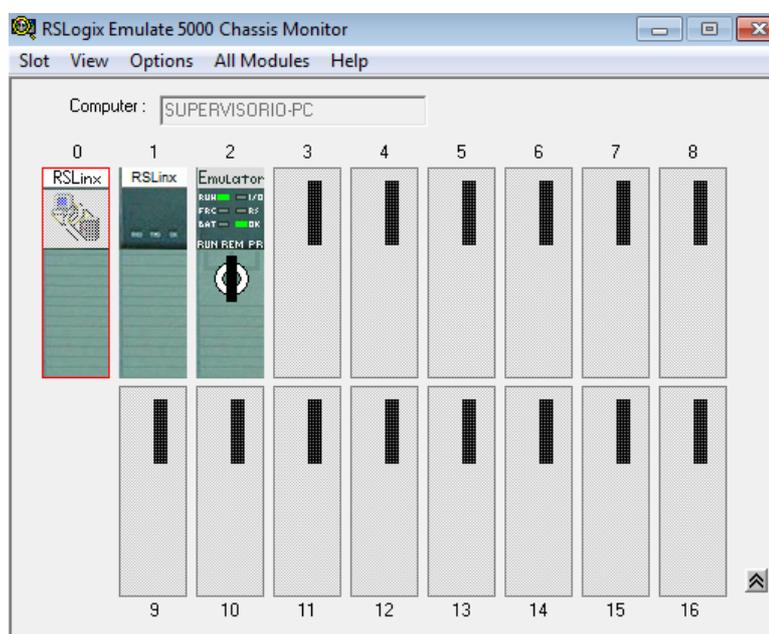
### 3.2.2. Equipamento de Controle

Um CLP tem a função de executar comandos programados, ler as entradas, acionar as saídas, trocar dados com outros controladores e comunicar-se com

<sup>2</sup> Disponível em: <<https://poi.apache.org/>>. Acesso em: 20 nov 2016.

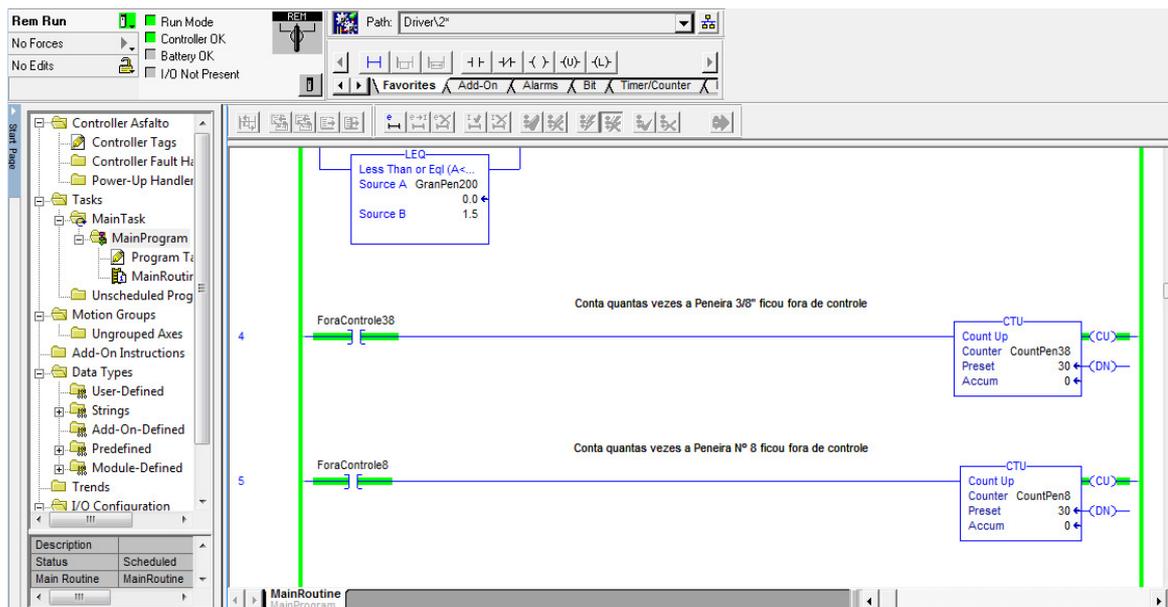
dispositivos remotos. A flexibilidade do *software* Ururau permite a comunicação com o CLP via OPC (OLE for Process Control).

Os testes foram realizados com o emulador de CLP RSEmulate 5000, da empresa Rockwell Automation. Este *software* permite emular um CLP pelo computador, sem necessidade de um CLP físico. Este recurso permite flexibilidade e redução nos custos na fase de testes e comissionamento. Sua interface é representada na Figura 4.



**Figura 4.** Interface do *Software* RSEmulate 5000.  
Fonte: Elaborado pelo Autor (2016).

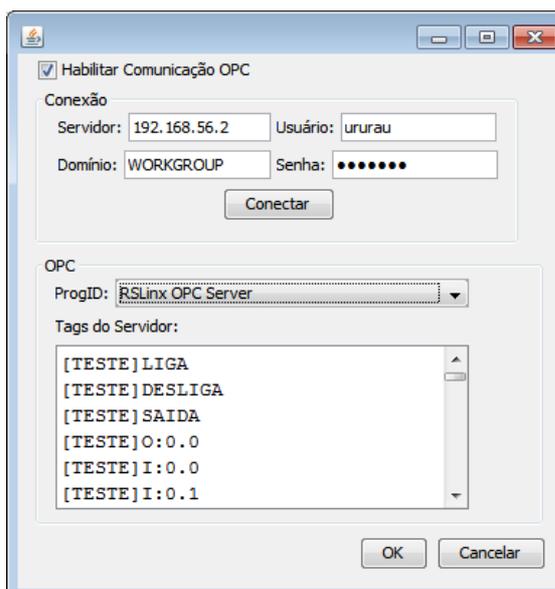
O *software* de programação utilizado foi o RSLogix 5000, onde será utilizado a programação na linguagem baseada no diagrama elétrico de relé (ladder), conforme ilustrado na Figura 5. A programação utilizada no trabalho está exposta em anexo no Apêndice B.



**Figura 5.** Programação em Ladder com RSLogix 5000.  
Fonte: Elaborado pelo Autor (2016).

### 3.2.4. Comunicação do Ururau com Controlador Lógico Programável

O Ururau permite a comunicação com o CLP. Esta funcionalidade pode ser acessada através da interface gráfica no menu “Ferramentas” pela opção “Comunicação OPC”. É necessário criar uma conta de usuário Windows com as permissões de acesso utilizadas para os componentes DCOM do servidor OPC escolhido. O procedimento completo para que a comunicação DCOM com o Ururau seja bem sucedida está em: <http://j-interop.dimentrix.com/quickstart.html>. A comunicação será bem sucedida quando clicar em ‘Conectar’ e a caixa de texto listar as tags pré-definidas no servidor OPC selecionado como mostra a Figura 6.



**Figura 6.** Comunicação OPC do Ururau  
Fonte: Elaborado pelo Autor (2016).

### 3.2.4. Ferramenta Solver do Excel

O Solver é uma ferramenta de análise incorporada no Microsoft Excel para Windows que foi desenvolvida para resolver equações não-lineares e é muito utilizada em programação matemática. Esta ferramenta possibilita a localização de um valor ideal para uma fórmula em uma célula chamada de célula destino na planilha. O Solver trabalha com um grupo de células relacionadas direta ou indiretamente com a fórmula na célula de destino. A ferramenta ajusta os valores nas células que o usuário configurar como células variáveis para produzir o resultado especificado pelo usuário na célula de destino. O usuário pode especificar as restrições do modelo, como maior ou igual ( $\geq$ ), menor ou igual ( $\leq$ ), igual ( $=$ ), dentre outros. A função objetivo também tem opções de ser maximizada, minimizada ou igualada a um valor desejado.

O uso da ferramenta é exemplificado na Figura 10, onde tem-se valores pré-definidos do processo (coluna A). As células variáveis (coluna B) foram restringidas para valores binários, pois podem ser utilizadas para ligar ou desligar algo no processo. O resultado da multiplicação das colunas A e B (coluna C) foi relacionado com a célula objetivo. O Solver tenta achar uma combinação dos valores de processo que quando ativados sua soma dê igual ao valor da célula objetivo. O

modelo de otimização utilizado no trabalho está explicado no Apêndice C.

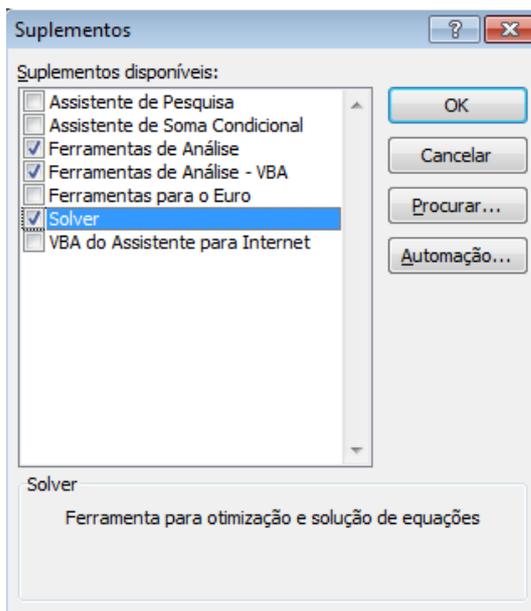
A	B	C	D
Valores de Processo	Células Variáveis	Valores de Processo x Células Variáveis	Célula Objetivo
30	0	0	50
30	0	0	
20	0	0	
10	1	10	
40	0	0	
30	1	30	
10	1	10	
5	0	0	
	SOMA	50	

**Figura 7.** Exemplo de aplicação da ferramenta Solver do Excel.  
Fonte: Elaborado pelo Autor (2016).

Para acionar a ferramenta Solver no Excel, os devidos passos deverão ser seguidos:

#### 3.2.4.1. Ativar o Suplemento Solver no Excel

Para isso deve-se abrir o menu “Opções do Excel”, ir na aba “Suplementos” e adicionar a ferramenta “Solver”, conforme ilustra a Figura 8.

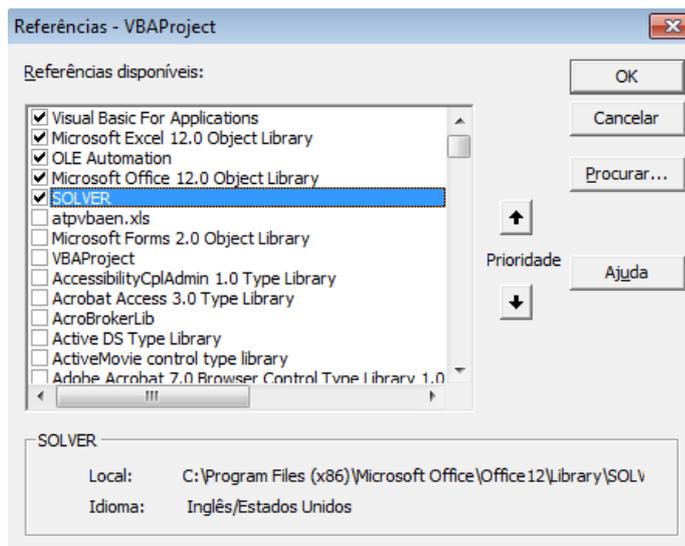


**Figura 8.** Ativação do Solver no Excel.  
Fonte: Elaborado pelo Autor (2016).

Após este passo o ícone do Solver do Excel aparecerá na aba “Dados” na guia “Análise”.

#### 3.2.4.2. Ativar a Referência de programação do Solver no VBA do Excel.

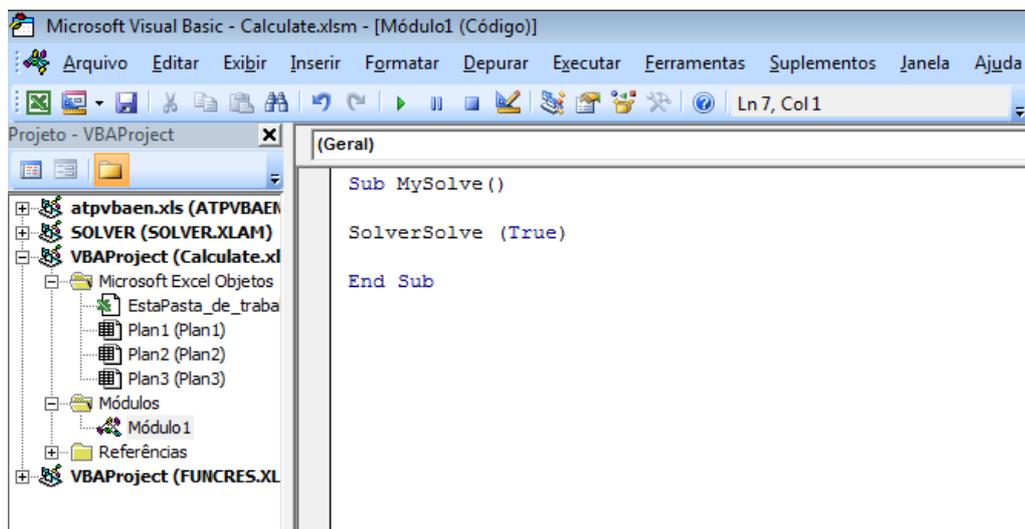
Deverá ser ativada a aba “Desenvolvedor” (cada versão de Excel tem uma maneira de ativá-la). Na aba “Desenvolvedor”, clicar no ícone “Visual Basic” ou pressionar ALT+F11. Com o VBA aberto, clique em Ferramentas > Referências e depois selecione a referência do Solver, conforme ilustra a Figura 9.



**Figura 9.** Ativação do Solver no VBA.  
Fonte: Elaborado pelo Autor (2016).

### 3.2.4.3. Inserir código VBA que aciona o Solver.

Crie um módulo de programação no VBA e insira o código SolverSolve (True), conforme Figura 10.



**Figura 10.** Programação VBA que aciona o Solver.  
Fonte: Elaborado pelo Autor (2016).

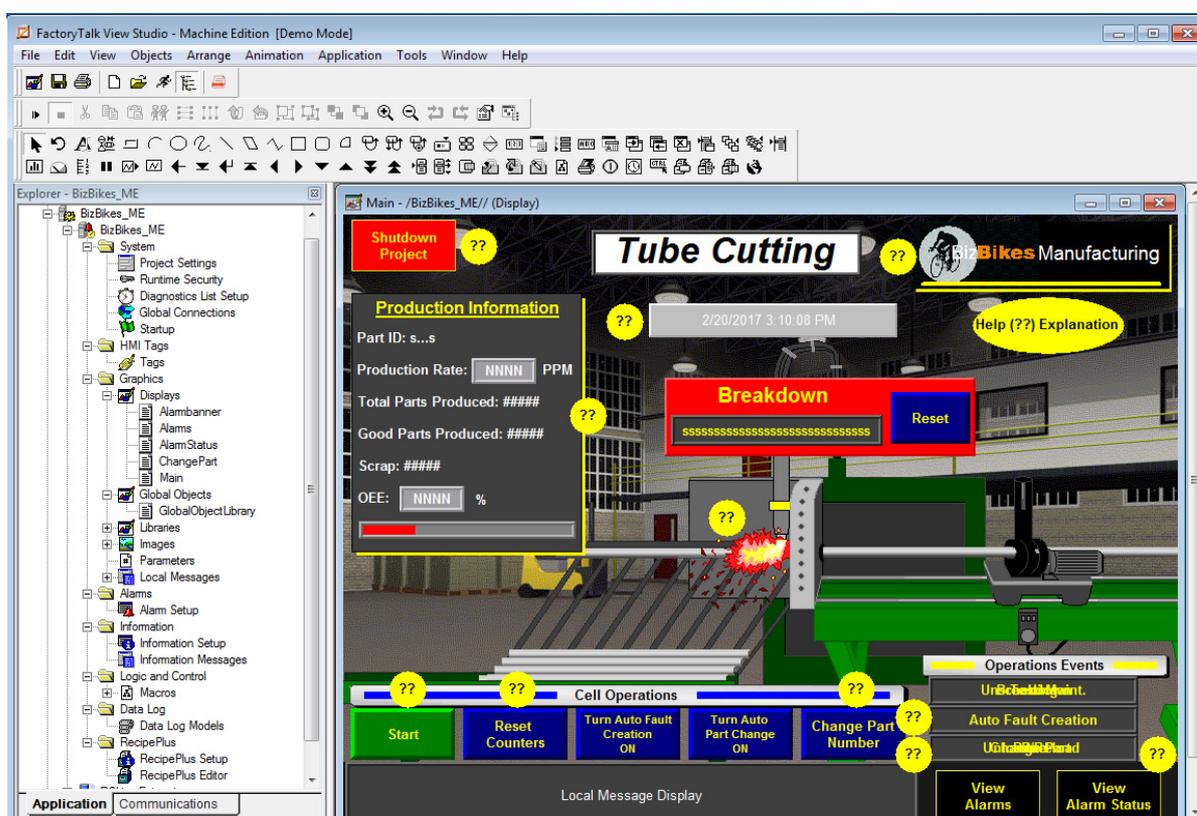
Desta forma o Excel está configurado para acionar o Solver.

### 3.2.5. Sistema de supervisão e aquisição de dados

O sistema de supervisão e aquisição de dados, também conhecido como SCADA ou sistema supervisorio, é utilizado para monitorar e supervisionar as variáveis de processo em tempo real. A operação do processo se torna muito mais dinâmica e rápida com este sistema, auxiliando no controle da qualidade do produto final.

O sistema supervisorio utilizado no trabalho foi o *FactoryTalk® View Studio* da empresa Rockwell Automation.

O processo da Usina Asfáltica foi representado e monitorado pelo sistema de supervisão e aquisição de dados, gerando gráficos e exibindo o momento em que o sistema está fora de controle. A ilustração do ambiente de desenvolvimento do *software Factory Talk View Studio* é demonstrada na Figura 11.



**Figura 11.** Ilustração do ambiente de desenvolvimento do *software Factory Talk View Studio*.  
Fonte: Elaborado pelo Autor (2016).

### 3.3. DESCRIÇÃO DO SISTEMA

#### 3.3.1. Processo de uma Usina de Asfalto

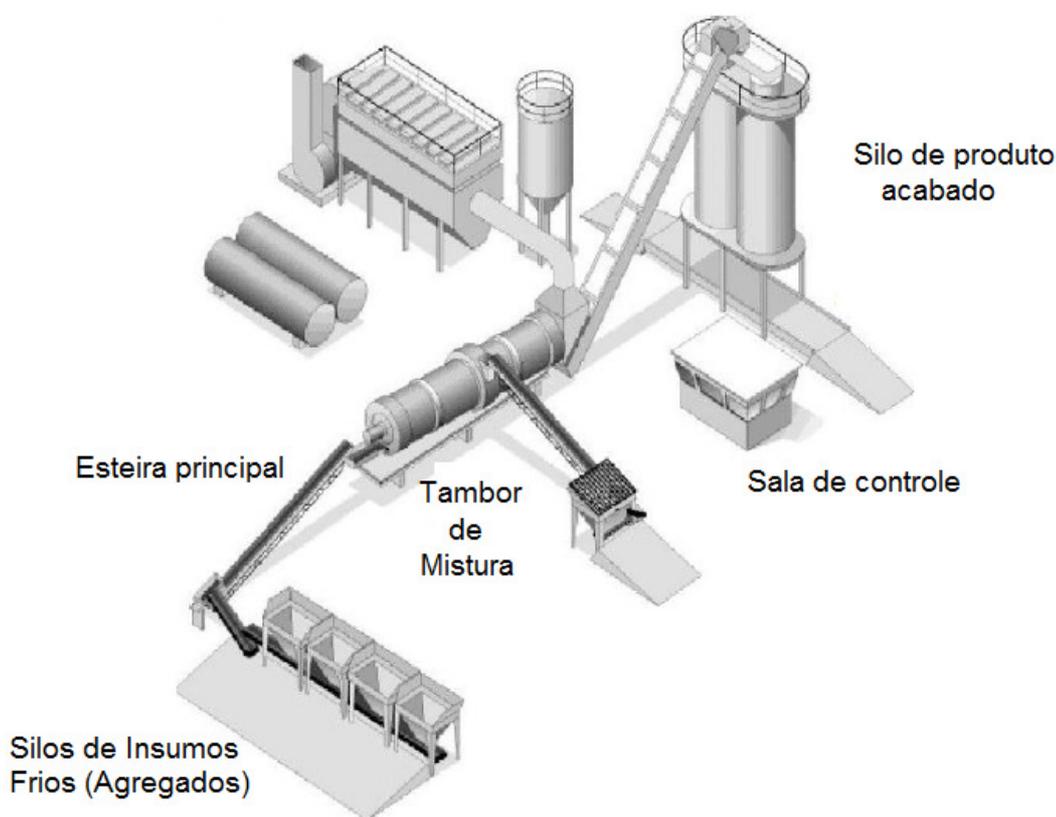
O processo que será utilizado como exemplo de aplicação do sistema de otimização será o processo de mistura asfáltica. Os processos de mistura em geral apresentam uma boa aplicabilidade para a solução apresentada neste trabalho.

A mistura asfáltica, representada na Figura 12, é uma combinação de agregados (britas de diversas granulometrias e areia) e asfalto. A combinação de agregados é de fundamental importância na qualidade do produto final da usina de asfalto, pois representa, em peso, cerca de 95% de uma mistura asfáltica enquanto o asfalto representa cerca de 5%. Já se tratando de volume, uma mistura asfáltica típica é composta por cerca de 85% de agregado, 10% de asfalto e 5% de vazios de ar (CHRISTENSEN, 2010).



**Figura 12.** Representação de uma mistura asfáltica típica.  
Fonte: Adaptado de Christense (2010)

Conforme exposto por Kabadurmus et. al. (2010), o processo típico de usina de asfalto é representado pela Figura 13, onde os agregados são armazenados nos silos de insumos frios e são liberados para serem utilizados no processo através da esteira transportadora principal em quantidades especificadas pelo projeto de mistura. A mistura de agregados é transportada para o tambor de mistura e secagem, onde a mistura é aquecida, adicionada emulsão asfáltica e outros componentes requeridos pelo processo. Por fim, a mistura asfáltica quente é armazenada nos silos de produto acabado para depois ser transportada para os locais de pavimentação.



**Figura 13.** Típica Usina de Asfalto.  
Fonte: Adaptado de Kabadurmus et. al. (2010).

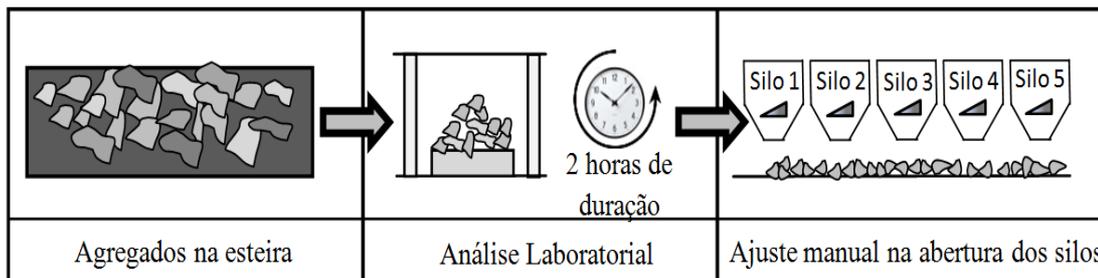
O produto final pode ter diversas especificações, de acordo com o solicitado pelos clientes. Por exemplo, asfaltos para estradas e regiões urbanas necessitam de especificações diferentes de projeto. Os principais aspectos para garantir uma boa qualidade do produto final são a qualidade do asfalto e a granulação dos agregados. A granulação é calculada como a porcentagem de produto que passa por cada

peneira. Os níveis de granulação dos agregados variam de fornecedor para fornecedor e também entre lotes do mesmo fornecedor.

O armazenamento dos agregados também influencia na variação da granulação, pois geralmente são armazenados em locais abertos e fatores climáticos, como umidade e chuva, afetam a especificação do insumo. Outro fator que interfere nas características dos agregados é o transporte do estoque (que geralmente são enormes) para os silos de insumos frios, que causa misturas heterogêneas e variação na granulação dos agregados.

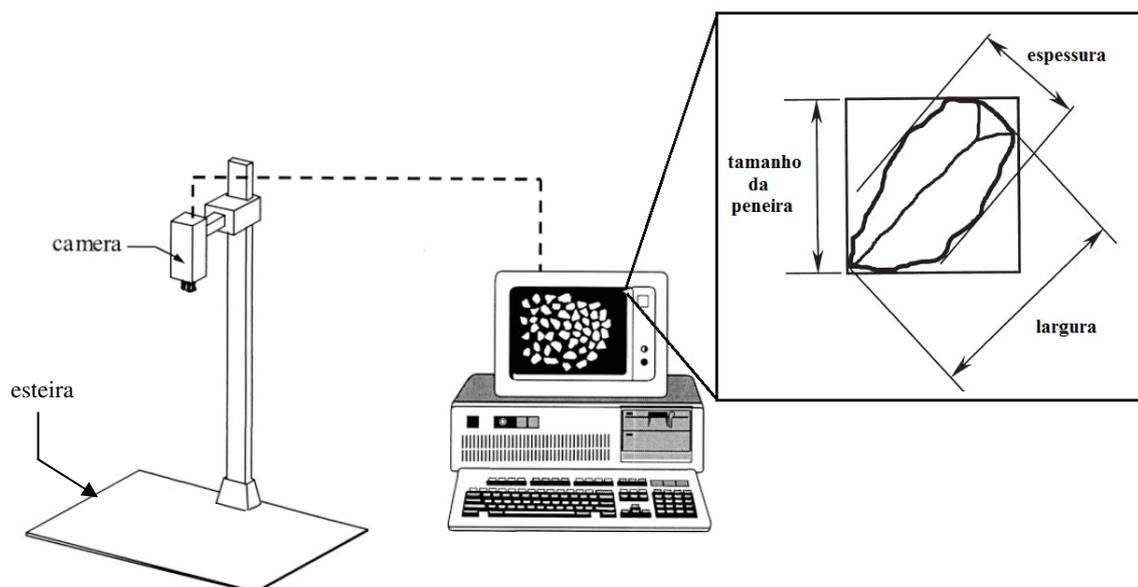
Normalmente, as pedreiras que fornecem os insumos para o processo não fragmentam os agregados antes de fornecê-los, entregam o produto na mesma condição em que são extraídos. Uma das soluções seria esmagar todos os agregados em porcentagens de granulações específicas utilizando máquina trituradora, porém isso acarretaria um grande aumento no custo e no tempo do processo. Diante destes fatos, a variação na granulação dos agregados continua sendo um problema na indústria do asfalto e a opção mais lógica é aceitar essa variação e aumentar a qualidade do processo usando outros métodos.

A prática comum de controle de qualidade na indústria de asfalto utiliza métodos off-line. A amostra é retirada do processo produtivo, e enviada para análise laboratorial para verificar se as características do produto estão de acordo com Projeto de Mistura (PM). Geralmente, o resultado da análise laboratorial demora cerca de 2 horas para ficar pronta. Se o resultado mostrar um problema com o produto, ações corretivas deverão ser tomadas por um operador. Porém, esta demora gera um produto com baixa qualidade e diversos custos relacionados inerentes. Considerando que um processo típico de produção de asfalto tem uma taxa de produção de 300 toneladas/hora, este método de controle de qualidade off-line pode gerar um desperdício de cerca de 600 toneladas de material (KABADURMUS, 2010). Ou ainda, este produto de má-qualidade pode ser utilizado nas rodovias e em pouco tempo se degradar. A Figura 14 demonstra de forma ilustrativa o fluxo típico de análises do processo de Usina Asfáltica.



**Figura 14.** Fluxo típico de análises do processo de Usina Asfáltica.  
Fonte: Elaborado pelo Autor (2016).

O método proposto é estimar constantemente os valores de granulação dos agregados através de um sistema de análise por imagem, demonstrado na Figura 15, e usar um modelo matemático de otimização da mistura que determine as proporções dos agregados a serem utilizados de cada silo de insumo frio. O sistema de análise por imagem aloca câmeras na esteira, verificando a granulação dos insumos de entrada e na saída do processo, verificando a granulação total da mistura asfáltica. As variações na qualidade do produto final da usina de asfalto podem ser minimizadas pela porcentagem ótima encontrada de cada silo de insumo frio utilizando um modelo de otimização de mistura para aumentar essa qualidade e padronização.



**Figura 15.** Configuração do sistema de análise de imagem.  
Fonte: Adaptado de Kwan (1999).

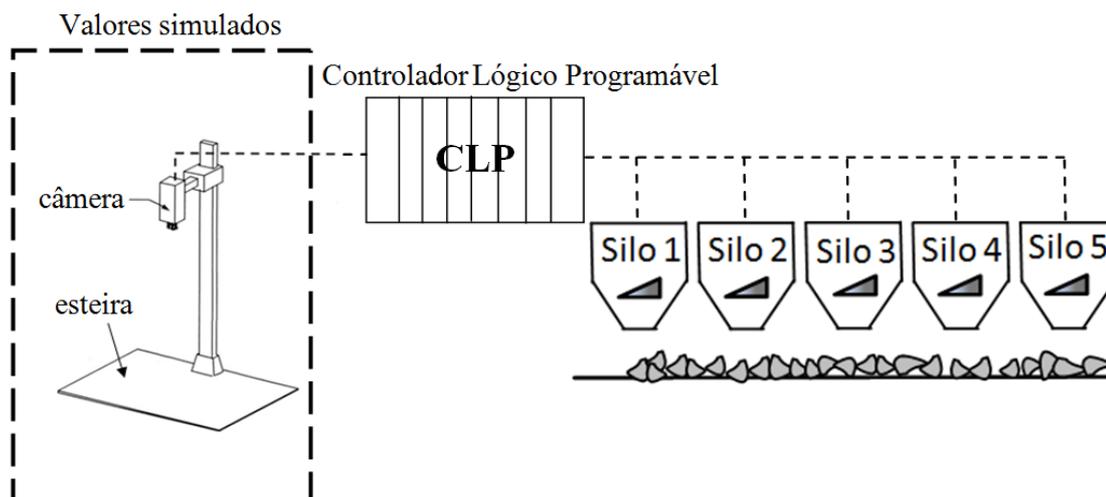
As imagens dos agregados são tiradas continuamente na esteira transportadora. As granulações dos agregados não poderão ser mudadas (elas são as entradas do processo) e terão grande variação. Se o nível de granulação do asfalto estiver fora dos limites de especificação como descrito no projeto de mistura, o sistema está fora de controle. Neste caso, ações corretivas serão tomadas para aumentar a qualidade e estabilidade do processo por meio da mudança das porcentagens dos agregados na mistura total. A melhoria na mistura será contínua para garantir uma melhor especificação do produto final mais perto possível do ideal. A mistura será alterada utilizando a abertura dos silos onde estão os agregados e a velocidade da esteira transportadora.

### **3.3.2. Sistema de Controle em Tempo Real**

O modelo de simulação online é elaborado no *software* Ururau e seu principal objetivo é mudar os parâmetros atuais do processo em tempo real. Como é de conhecimento, mudar os parâmetros no processo real pode ser muito custoso e a simulação permite que o processo seja testado usando diferentes cenários, possibilitando ajustes no controle e análise profunda do processo.

No sistema utilizado neste trabalho, os valores de granulação dos agregados são simulados através de um *software* de simulação de eventos discretos, com valores hipotéticos de média e desvio padrão relatados na Tabela 2. Estes valores são enviados para uma planilha (explicada nas próximas seções) onde é calculada a proporção ótima de combinação dos agregados. O CLP recebe estes valores e envia os sinais de abertura e fechamento dos silos, conforme ilustra a Figura 16.

Desta forma, pode-se realizar um controle do processo em tempo real, monitorando continuamente o parâmetro do processo a ser controlado e agindo corretivamente, quando necessário, nos atuadores que modificam as características do processo.

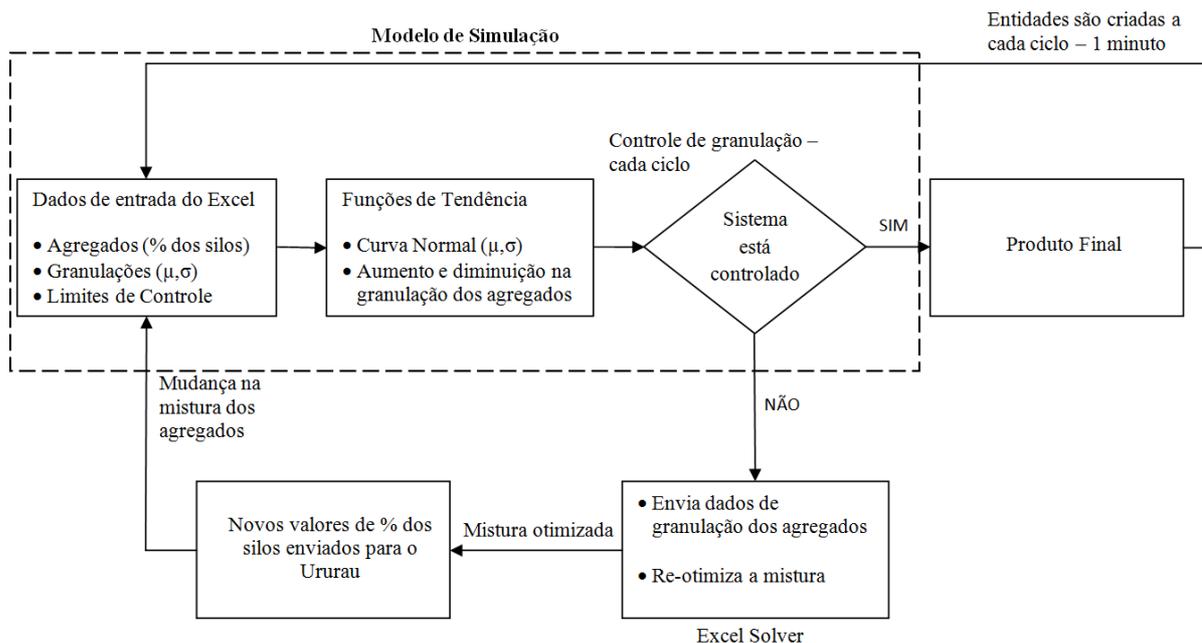


**Figura 16.** Configuração do sistema de controle por análise de imagem.  
Fonte: Elaborado pelo Autor (2016).

Observando a Figura 16, pode ser notado na área em destaque que os valores de granulação dos agregados podem ser simulados através de um *software* de SED. A simulação destes valores nos permite realizar testes em diferentes cenários, dando maior flexibilidade ao sistema.

O processo real da usina de asfalto é um processo contínuo, porém a simulação do sistema é feita em um *software* de simulação discreta. Portanto, é necessário utilizar o recurso de discretização de unidades. A discretização funciona como pedaços de agregados discretizados que são criados individualmente como entidades para serem combinadas e construídas como pedaços discretizados de produto final.

Conforme demonstrado na Figura 17, primeiramente os valores de Limites de Controle, Funções de tendência de variação dos agregados e porcentagem de abertura dos silos são lidos de um arquivo do Excel. Então, o modelo de simulação inicia, e neste ponto a função de tendência começa a trabalhar. As funções de tendência são ferramentas importantes na simulação, pois auxiliam o ambiente computacional simular as condições do mundo real. Foi utilizada a curva Normal para simular o processo real, com valores determinados de média e desvio padrão que serão especificados em seção subsequente.



**Figura 17.** Integração da modelagem de simulação com o Excel Solver  
 Fonte: Adaptado de Kabadurmus *et. al.* (2010).

O sistema verifica constantemente se o sistema está controlado. Assume-se como sistema fora de controle quando a granulação de uma das peneiras estiver fora dos limites de especificação do PM. À medida que as entidades são criadas de acordo com o intervalo de tempo de discretização, o produto final também é verificado se está dentro dos parâmetros de controle de acordo com este tempo pré-definido. O intervalo de tempo em que as entidades são criadas na simulação é de 1 minuto.

Se o bloco do produto final estiver dentro dos parâmetros de controle do processo, então não é necessário mudar a mistura. Porém, se o produto final estiver fora das especificações, ações corretivas são tomadas. Estas ações corretivas são consideradas como as mudanças nas porcentagens de abertura dos silos de agregados afim re-otimizar a mistura.

Como entrada para o sistema de otimização, os valores das granulações atuais de cada peneira são lidas do Ururau e escritas no Excel. Após o modelo de otimização re-otimizar a mistura dando novos valores de porcentagem aos agregados, estes valores são lidos do arquivo Excel e escritos de volta para o Ururau, e a simulação continua com os novos valores da mistura. A simulação continua até atingir o valor estipulado de 10 horas, que é um dia de produção.

### 3.3.3. Dados de Entrada e Parâmetros

Em paralelo, o modelo funciona com uma planilha em um arquivo do MS Excel. Neste arquivo, estão contidas as entradas (*inputs*) do modelo de simulação. Os dados de entrada para teste do sistema é provido pelo Centro Nacional de Tecnologia Asfáltica dos EUA, reportado no artigo de Zhang (2014). Estes dados são hipotéticos e observados comumente em operações de usinas de asfalto, assim como as especificações e limites de projetos de misturas típicos utilizados.

Os dados são compostos por 3 partes (conforme Tabelas 2 e 3). A primeira parte é a média e o desvio padrão da granulação de cada peneira para os 5 silos. Uma simplificação do processo será a consideração de apenas as peneiras mais influentes ao invés de todas 8 peneiras que são utilizadas comumente. As 4 peneiras mais influentes são as de 3/8", nº 8, nº 30 e nº 200. Os valores de média e desvio padrão da granulação das peneiras são expressos em porcentagem de material que passará por cada uma delas e são mostrados na parte superior da Tabela 2. A segunda parte é o Projeto de Mistura e os limites superiores e inferiores de tolerância para cada especificação ou produção, que são mostrados nas últimas linhas da Tabela 2 e indicam valores de tolerância para mais ou para menos ( $\pm$ ) toleráveis da meta de Projeto de Mistura.

**Tabela 2.** Dados de entrada e parâmetros dos agregados

Critério de Agregados	SILOS	PENEIRAS			
		3/8"	8	30	200
<b>Média das granulações</b>	1	52	14	7,9	3,7
	2	100	96	51	14
	3	100	86	41	1,4
	4	100	7	1	0,1
	5	100	52	9	0,7
<b>Desvio padrão das granulações</b>	1	3,00	2,00	1,00	1,00
	2	0,00	1,00	4,00	2,00
	3	0,00	2,00	3,00	0,50
	4	0,00	1,00	0,50	0,50
	5	0,00	3,00	2,00	0,01
<b>Meta de porcentagem das peneiras (PM)</b>		82,20	33,00	14,30	3,50
<b>Limites de tolerância da especificação</b>		8,00	5,00	4,00	2,00
<b>Limites de tolerância da produção</b>		5,00	4,00	3,00	2,00

Fonte: Zhang (2014).

As porcentagens iniciais do Projeto de Mistura e os limites superiores e inferiores dos 5 silos de insumos frios são demonstrados na Tabela 3. A porcentagem de cada silo durante a produção deve estar dentro dos seus limites superior e inferior, e isto restringe o procedimento de otimização de processo.

**Tabela 3.** Dados iniciais e limites de abertura dos silos

Silos	Proporção Inicial	LIMITE PERCENTUAL DOS SILOS	
		Mínimo	Máximo
1	37	17	57
2	14	5	23
3	7	2	12
4	30	10	50
5	12	5	23

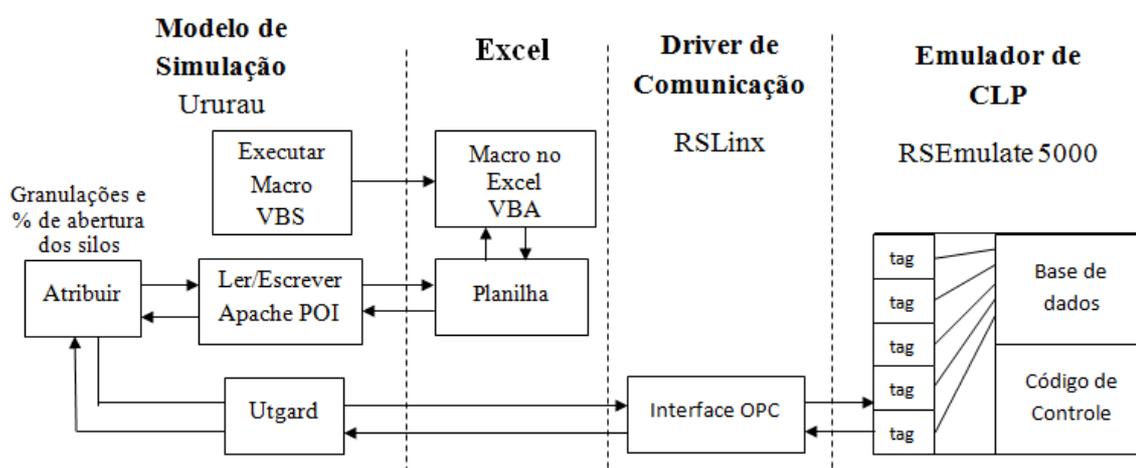
Fonte: Zhang (2014).

### 3.3.4. Integração do modelo de simulação com o CLP e o Excel

O *software* Ururau tem a capacidade de trocar informações com CLPs e com o Excel. Esta interatividade nos permite uma maior flexibilidade no sistema, podendo utilizar a simulação para auxiliar nos testes de outros sistemas.

O Ururau troca informações com o Excel utilizando o Java API (Application Programming Interface) Apache POI<sup>3</sup> que manipula vários formatos de arquivos baseados nos padrões Office Open XML e Microsoft's OLE 2 Compound Document format (OLE2).

Observe na Figura 18 que a planilha se comunica diretamente com o *software* Ururau, utilizando bloco “Ler/Escrever” que pode armazenar as informações do Excel como variáveis internas do Ururau. Este *software* de simulação também tem a capacidade de executar macros VBA contidos na planilha. Todas as informações contidas no Excel podem ser enviadas para o CLP através do bloco “Atribuir” que utiliza a Interface OPC para alterar os valores na base de dados do CLP.

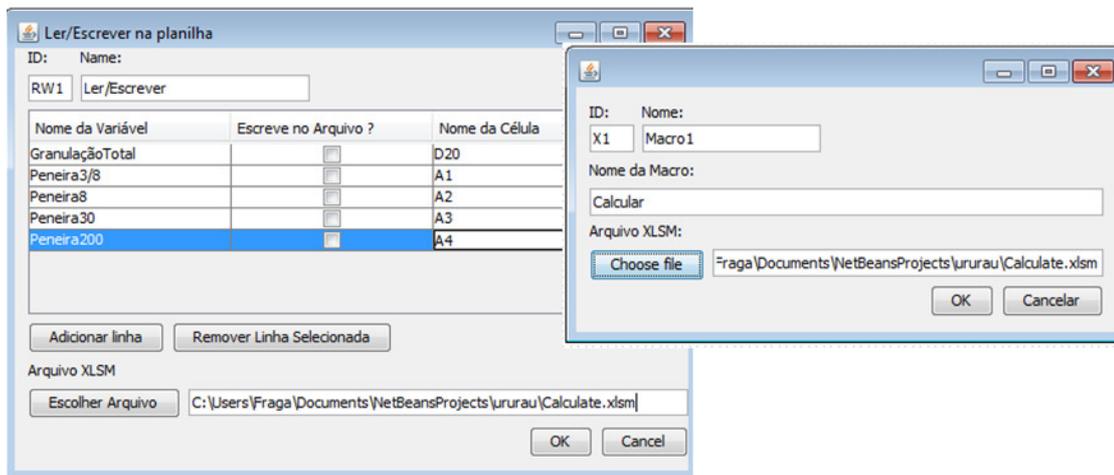


**Figura 18.** Integração do modelo de simulação, Ururau, Excel e CLP.  
Fonte: Elaborado pelo Autor (2017).

Foi desenvolvido o bloco “Ler/Escrever” e um bloco “Executar Macro” no ambiente de desenvolvimento do Ururau para que o usuário possa ler e escrever

<sup>3</sup> Disponível em: < <https://poi.apache.org/>>. Acesso em: 20 jan 2017.

informações nas células do Excel e rodar macros em código VBA (Visual Basic for Applications). Estes blocos são representados pela Figura 19.



**Figura 19.** Ler/Escriver e Executar Macro na Interface do Ururau  
Fonte: Adaptado de Kabadurmus *et. al.*; 2010

O bloco “Ler/Escriver” serve para ler valores da planilha e armazená-los como variáveis no Ururau ou escrever variáveis do Ururau em células do Excel. O usuário deve definir os nomes das variáveis, selecionar se a variável será lida ou escrita no Excel (se a caixa presente na coluna “Escrever no Arquivo?” estiver desmarcada, o Ururau pode ler o valor da célula e se estiver marcada, o Ururau pode escrever na célula), definir a célula da planilha que será manipulada e informar o caminho onde o arquivo está contido no computador. No bloco de comando “Executar Macro” o usuário deve informar o nome da macro desenvolvida no Excel e o caminho onde o arquivo está contido no computador.

Ainda conforme Figura 18, para que as macros em VBA sejam executadas, foi utilizado um arquivo em código VBS (Visual Basic Script) parametrizado que é executado pelo Ururau em Java e quando este arquivo VBS é acionado o Excel é aberto, executa a macro desejada e salva o arquivo guardando os valores atualizados.

As informações da simulação são enviadas para o CLP utilizando o bloco “Atribuir” através da integração do Ururau com o servidor OPC com o cliente OPC

Utgard<sup>4</sup>. Este cliente OPC faz parte do *software* OpenSCADA porém pode ser utilizado independentemente da plataforma. O Utgard é um Cliente API Java OPC 100% puro e gratuito.

Foi utilizado o emulador da Rockwell Automation RSEmulate 5000 que simula o comportamento de um CLP real e comunica com o Ururau através do seu Driver de comunicação OPC RSLinx.

### 3.4. DESCRIÇÃO DOS CENÁRIOS

Os experimentos foram realizados em cenários diferentes, com o objetivo de testar a solução com Solver em diversas situações do processo e verificar em qual ela tem o melhor resultado. Essa variação no cenário consiste basicamente em variar o comportamento dos insumos do processo e variar os limites de controle considerados.

Foram utilizadas duas situações de variação no comportamento dos insumos, o de “baixa variação” e “alta variação”, descritos na Tabela 4.

---

<sup>4</sup> Disponível em: < [http:// openscada.org/projects/utgard](http://openscada.org/projects/utgard) >. Acesso em: 20 jan 2017.

**Tabela 4.** Dados de Baixa Variação e Alta Variação nos insumos

		<b>BAIXA VARIAÇÃO</b>			
		Peneiras			
Critério de Agregados	Silos	3/8"	Número 8	Número 30	Número 200
Média das granulações	1	52	14	7,9	3,7
	2	100	96	51	14
	3	100	86	41	1,4
	4	100	7	1	0,1
	5	100	52	9	0,7
Desvio padrão das granulações	1	3	2	1	1
	2	0	1	4	2
	3	0	2	3	0,5
	4	0	1	0,5	0,5
	5	0	3	2	0,01
		<b>ALTA VARIAÇÃO</b>			
		Peneiras			
Critério de Agregados	Silos	3/8"	Número 8	Número 30	Número 200
Média das granulações	1	52	14	7,9	3,7
	2	100	96	51	14
	3	100	86	41	1,4
	4	100	7	1	0,1
	5	100	52	9	0,7
Desvio padrão das granulações	1	4,5	3	1,5	1,5
	2	0	1,5	6	3
	3	0	3	4,5	0,75
	4	0	1,5	0,75	0,75
	5	0	4,5	3	0,015

Fonte: Elaborado pelo Autor (2017).

Note na Tabela 4 que os valores de média das granulações nas duas situações, se mantêm os mesmos. Porém, os valores de desvio padrão das granulações se alteram. Os valores da Alta Variação são 1,5 vezes superior aos valores de Baixa Variação. Portanto a oscilação nos insumos de entrada do processo será maior quando considerada os valores de Alta Variação. Já a variação nos limites de controle, serão considerados duas situações, Limite de Tolerância da Especificação e Limite de Tolerância da Produção, descritos na Tabela 5.

**Tabela 5.** Dados dos limites de controle utilizados.

	<b>PENEIRAS</b>			
	3/8"	Número 8	Número 30	Número 200
<b>Meta de porcentagem das peneiras (PM)</b>	82,2	33	14,3	3,5
<b>Limites de tolerância da especificação</b>	8	5	4	2
<b>Limites de tolerância da produção</b>	5	4	3	2

Fonte: Elaborado pelo Autor (2017).

Observado a Tabela 5, podemos notar que os Limites de Tolerância da Especificação são maiores do que os Limites de Tolerância na Produção, permitindo que as variáveis do processo oscilem mais, sem ser considerado fora de controle.

Os experimentos foram realizados, combinando todas estas diferentes situações, gerando na tabela 6.

**Tabela 6.** Combinação dos cenários

<b>Limite de Controle</b>	<b>Varição nos Insumos</b>
LTE	Baixa
LTE	Alta
LTP	Baixa
LTP	Alta

Fonte: Elaborado pelo Autor (2017).

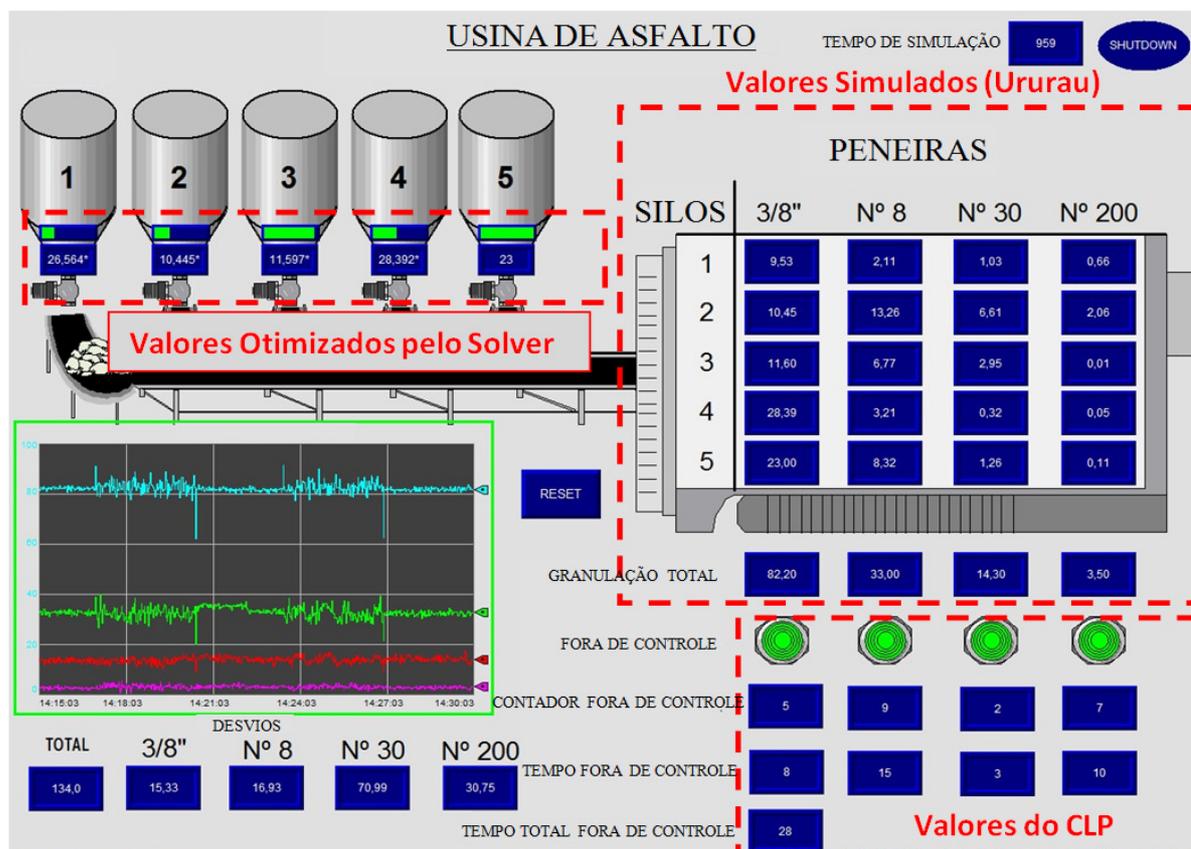
De acordo com a tabela 6, ao todo foram gerados 4 cenários para realizar testes no sistema.

## 4. RESULTADOS E DISCUSSÃO

### 4.1. MODELO DE SIMULAÇÃO

A simulação foi utilizada para variar os valores da granulação dos agregados de acordo com o sistema real, que permite testes e comissionamento do sistema com grande economia e flexibilidade. Mudar os parâmetros no processo real pode ser muito custoso e a simulação permite que o processo seja testado usando diferentes cenários, possibilitando ajustes no controle e análise profunda do processo. Neste trabalho, foi utilizado o *software* de SED, chamado de Ururau. O *software* foi utilizado para gerar as curvas de tendência da granulação dos agregados, enviar informações para o Excel e se comunicar com o CLP e com o sistema supervisor.

A simulação pode ser acompanhada em tempo real pelo sistema supervisor que foi desenvolvido no *Factory Talk View SE* da empresa Rockwell Automation, gerando gráficos em tempo real e possibilitando comandos como Reset de contadores, e é ilustrado na Figura 20. Todos os parâmetros da simulação estão descritos no Apêndice A.



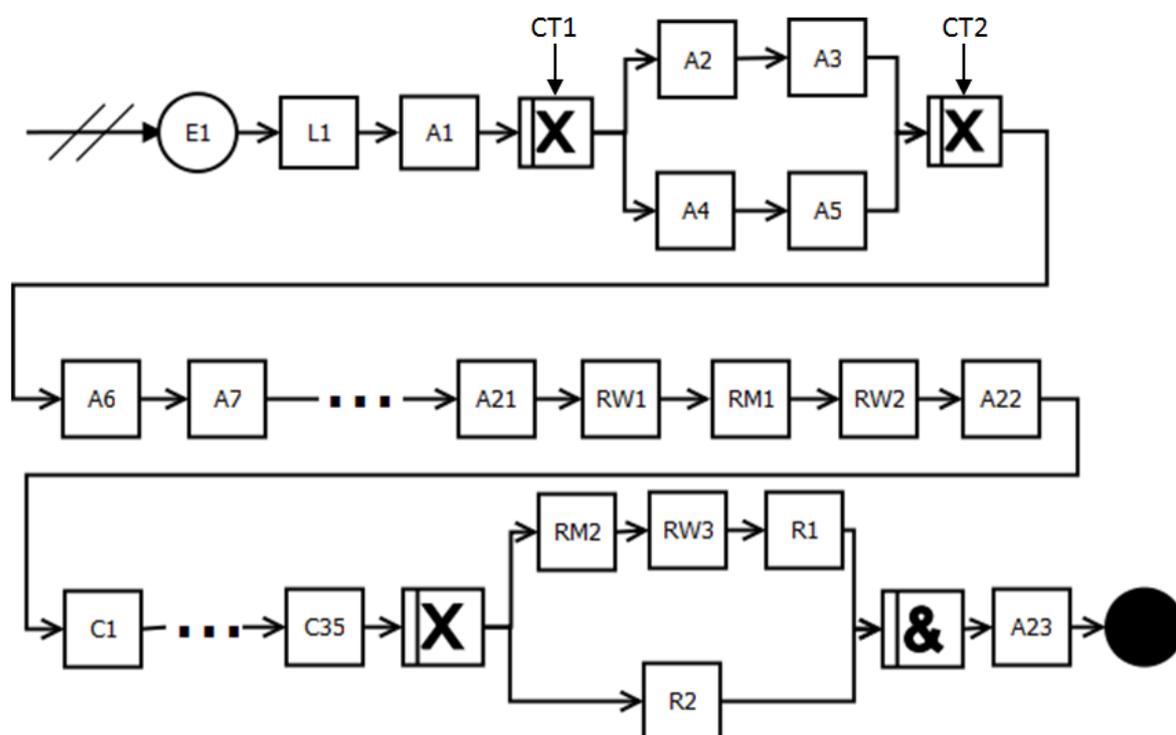
**Figura 20.** Ilustração da Interface Homem-Máquina da Usina de Asfalto  
Fonte: Elaborado pelo Autor (2017).

Conforme ilustrado na Figura 20, os valores simulados são representados como valores individuais de granulação de cada tipo de produto oriundo dos silos em cada peneira. Os valores de granulação total também são representados numericamente e é gerado um gráfico com esses valores para acompanhamento do histórico do comportamento do processo.

Quando alguma peneira está fora de controle, a luz de sinalização “Fora de Controle” ficará da cor vermelha. Esta indicação é controlada pelo CLP com blocos de comparação que monitoram se os valores estão dentro dos parâmetros de controle. Assim como os contadores “Contador Fora de Controle” (que conta quantas vezes cada peneira ficou fora de controle), “Tempo Fora de Controle” (que conta o tempo que cada peneira ficou fora de controle) e “Tempo Total Fora de Controle” (que conta o tempo total que o sistema ficou fora de controle) que são monitorados em tempo real pelo CLP. As indicações com a porcentagem de abertura de cada silo e seus respectivos valores são gerados pela otimização do

Solver que calcula a porcentagem ótima de abertura de cada silo quando o sistema está fora de controle.

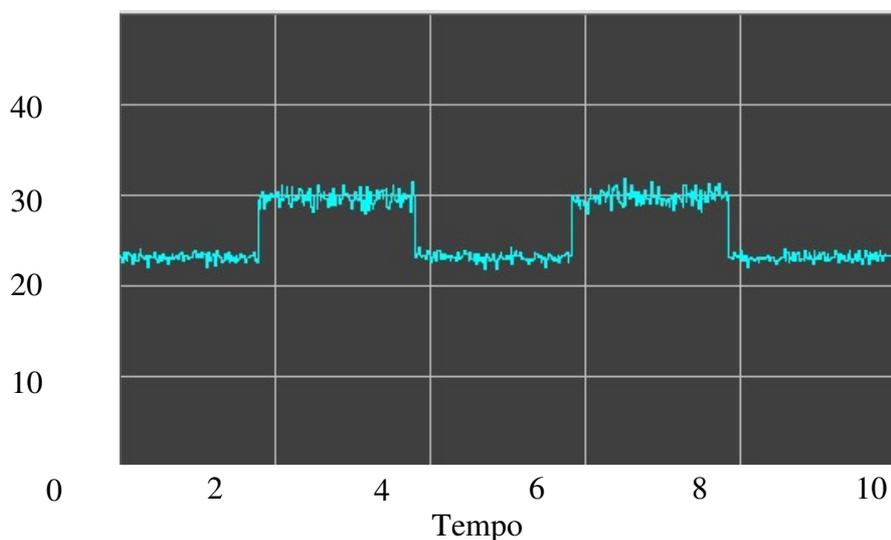
O modelo conceitual do sistema é baseado em IDEF-SIM que é uma linguagem utilizada para a descrição conceitual de um modelo de simulação, apresentada por Montevechi et al. (2010). Assim, a utilização deste simulador possui uma curva de aprendizagem menor, quando comparado com os demais, pois há uma proximidade entre o modelo conceitual e o modelo computacional. O modelo conceitual utilizado para simular a Usina de Asfalto é representado pela Figura 21.



**Figura 21.** Modelo conceitual em IDEF-SIM.  
Fonte: Elaborado pelo Autor (2017).

O elemento E1 (Criar) cria entidades a cada 1 minuto na simulação. Os elementos L1 (Segurar), A1 (Atribuir) e A23 (Atribuir), formam um conjunto onde a próxima entidade criada pelo elemento E1 só é liberada após a entidade anterior ter passado por todos os elementos da simulação. Ou seja, as entidades percorrem os elementos da simulação, uma de cada vez. A estrutura de OR, A2, A3, A4 e A5 (Atribuir), é utilizada para gerar uma função de tendência que descreve uma possível mudança na granulação dos agregados ao longo do tempo. A função de tendência

utilizada no trabalho é a função de passo alternado, conforme ilustra o Gráfico 5.



**Gráfico 5.** Função de passo alternado.  
Fonte: Elaborado pelo Autor (2017).

A função de tendência aumenta e diminui no intervalo de cada duas horas simuladas em 33,33% as variáveis ao longo do tempo durante a simulação em torno de sua média. Os elementos de A6 até A21 (Atribuir) geram as curvas normais de cada granulação dos agregados, multiplicados pela função de tendência. O elemento RW1 (Ler/Escriver no Excel) escreve no Excel os valores de granulação gerados. O RM1 (Executar Macro no Excel) aciona a macro que irá atualizar os cálculos do Excel e depois o RW2 irá ler os valores atualizados e calculados da planilha.

O A22 tem a função de totalizar os desvios das peneiras. A estrutura C1 até C35 tem a função de comunicar todos os valores necessários com o CLP. A estrutura OR, RM2, RW3, R1 e R2 tem a função de verificar se tem alguma peneira fora de controle. Se existir uma ou mais fora de controle, então o RM2 irá acionar o Solver que irá calcular as proporções de abertura dos silos para otimizar a mistura. Estes valores serão lidos do Excel pelo elemento RW3 e o R1 irá contar quantas vezes a mistura foi otimizada. O elemento R2 irá contar quantas vezes a mistura estava dentro do controle e não foi necessária a otimização.

## 4.2. MODELO DE OTIMIZAÇÃO

Quando o sistema é detectado que está fora de controle, o processo de otimização é iniciado. Os dados do processo simulados pelo Ururau são enviados para uma planilha no Excel e então o aplicativo Solver é iniciado para encontrar a solução ótima para o sistema. O objetivo da otimização é minimizar o desvio total entre a granulação da mistura e a granulação prevista no Projeto de Mistura, em todas as peneiras. Todas as restrições do processo dependem do Projeto de Mistura requerido. O algoritmo de otimização é definido a seguir:

$$\min \sum_j D_j = \frac{|n_j - \sum_i x_i g_{ij}|}{(r_j^{\max} - r_j^{\min})/2}$$

**Equação 1.** Definição do algoritmo de otimização (1)  
Fonte: Elaborado pelo Autor (2017)

**s. a.**

$$s_i^{\max} \leq x_i \leq s_i^{\min}$$

**Equação 2.** Definição do algoritmo de otimização (2)  
Fonte: Elaborado pelo Autor (2017)

$$r_j^{\max} \leq \sum_i x_i g_{ij} \leq r_j^{\min}$$

**Equação 3.** Definição do algoritmo de otimização (3)  
Fonte: Elaborado pelo Autor (2017)

$$\sum_i x_i = 1$$

**Equação 4.** Definição do algoritmo de otimização (4)  
Fonte: Elaborado pelo Autor (2017)

$$x_i \geq 0$$

**Equação 5.** Definição do algoritmo de otimização (4)  
 Fonte: Elaborado pelo Autor (2017)

Onde  $i$  é cada silo de insumo frio,  $j$  é cada peneira,  $D_j$  é o desvio total.

A variável de decisão do modelo é:

#### 4.2.1. Porcentagem do peso da mistura total vindo dos silos ( $x_i$ ).

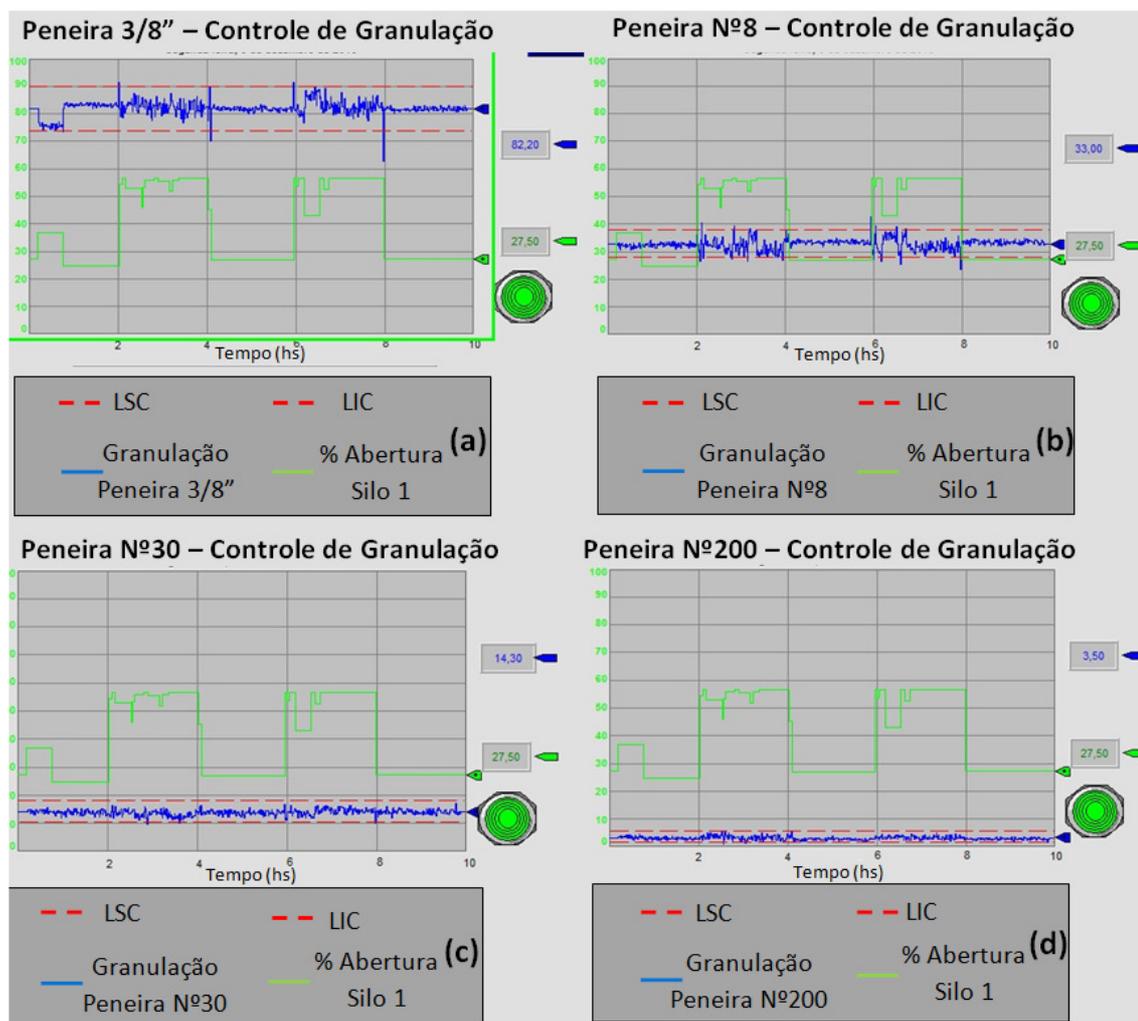
Os parâmetros são:

- (1). Granulação dos agregados medidas nos silos ( $g_{ij}$ );
- (2). Meta % (do Projeto de Mistura) de passagem nas peneiras ( $n_j$ );
- (3). Limite superior e inferior de especificação % de passagem nas peneiras ( $r_j^{max}, r_j^{min}$ );
- (4). Limite superior e inferior para % de peso vindo de cada silo ( $s_i^{max}, s_i^{min}$ );

#### 4.3. EXPERIMENTOS SIMULADOS

Os resultados foram obtidos através de uma única rodada de simulação equivalentes há 10 horas (600 minutos) simuladas, correspondendo a um dia de trabalho em uma típica usina de asfalto. Estes parâmetros foram utilizados em cada cenário testado, utilizando uma replicação para cada cenário.

Os resultados são exibidos no *software* da IHM gerando gráficos e exibindo os valores instantâneos das variáveis, conforme observado na Figura 22.



**Figura 22.** Captura de tela dos gráficos da simulação para Baixa Variação e com a solução do Solver ativa<sup>5</sup>

Fonte: Elaborado pelo Autor (2017)

Conforme demonstrado na Figura 22, os valores de granulação de cada peneira podem ser observados historicamente através de gráficos gerados pelo sistema supervisor. Este recurso nos permite observar o comportamento do sistema ao longo do tempo e todas as mudanças de abertura dos silos conforme o sistema fica fora de controle.

Também pode-se observar os resultados pelo relatório gerado pelo *software* de simulação Ururau, que ao final da rodada de simulação, gera um relatório detalhado da simulação e suas variáveis. Um exemplo de uma parte do relatório é

<sup>5</sup> : (a) granulação da peneira 3/8" para todos os silos (acima) e porcentagem de abertura do silo 1 (abaixo); (b) granulação da peneira N°8 para todos os silos e porcentagem de abertura do silo 1; (c) granulação da peneira N°30 para todos os silos (abaixo) e porcentagem de abertura do silo 1 (acima); (d) granulação da peneira N°200 para todos os silos (abaixo) e porcentagem de abertura do silo 1

representado pela Figura 23.

<b>User Variables</b>	mean	standard deviation	half-width	minimum	maximum
GradSieve38	82.370	NaN	NaN	82.370	82.370
GradSieve8	34.330	NaN	NaN	34.330	34.330
GradSieve30	13.156	NaN	NaN	13.156	13.156
GradSieve200	3.249	NaN	NaN	3.249	3.249
TotalcumDeviation	572.056	NaN	NaN	572.056	572.056

<b>Counters</b>	mean	standard deviation	half-width	minimum	maximum
R1 CounterSolver	25.000	NaN	NaN	25.000	25.000
R2 CounterOK	576.000	NaN	NaN	576.000	576.000

**Figura 23.** Parte do relatório gerado pelo Ururau.  
Fonte: Elaborado pelo Autor (2017)

Note que na Figura 23, o relatório exibido pelo Ururau também tem informações importantes, como as médias das granulações de cada peneira, o desvio total acumulado e contadores de quantos ciclos da simulação o solver foi iniciado ou não. Como a simulação foi feita com apenas uma replicação, alguns valores do relatório não apresentam valores, pois precisam de mais de uma replicação para serem gerados. A análise dos resultados é baseada no tempo fora de controle e no desvio total acumulado. O tempo fora de controle corresponde a quanto tempo simulado em minutos em que o sistema ficou fora do Limite de Controle onde o máximo é 600 minutos (10 horas de tempo simulado). O valor do desvio total acumulado é calculado no *software* Ururau, onde a cada ciclo da simulação (a cada 1 minuto simulado) o Ururau obtém o valor do somatório das porcentagens de desvio de cada peneira em relação a meta da PM (equação 1) e vai totalizando.

Os resultados estão resumidos na Tabela 7.

**Tabela 7.** Resultados das simulações

Limite de Controle	Variação nos Insumos	Estratégia de Controle		% Redução	
		Sem Controle	Controle Solver	Tempo fora de Controle <sup>6</sup>	Desvio Total Acumulado
LTE <sup>7</sup>	Baixa	31 (0) 804,86	25 (25) 572,05	19,35	28,93
LTE <sup>8</sup>	Alta	70 (0) 944,22	63 (67) 735,95	10,00	22,06
LTP	Baixa	544 (0) 1156,73	40 (39) 710,81	92,65	38,55
LTP	Alta	502 (0) 1313,23	139(140) 934,62	72,31	28,83

Fonte: Elaborado pelo Autor (2017)

Os valores numéricos obtidos na coluna “Estratégia de Controle” correspondem ao tempo total em que o sistema ficou fora de controle (número contido na parte superior à esquerda de cada célula), quantas vezes o solver foi iniciado (número entre parênteses na parte superior à direita de cada célula) e o desvio total acumulado das peneiras (número contido na parte inferior de cada célula). Já na coluna “% Redução” os valores numéricos obtidos correspondem a porcentagem de redução no tempo fora de controle e no desvio total acumulado de cada cenário, comparando o sistema sem nenhuma estratégia de controle com o sistema sendo controlado pelo Solver.

Conforme observado na Tabela 6, a otimização obteve um melhor resultado com o Limite de Tolerância de Produção (LTP), que é mais restrito do que o Limite de Tolerância de Especificação (LTE), apresentando uma melhoria de 92.65% tempo fora de controle comparando com sistema sem controle pelo Solver. Porém, a otimização não se mostrou tão eficaz quando o sistema apresentou uma alta

<sup>6</sup> LTE (Limite de Tolerância de Especificação); LTP (Limite de Tolerância de Produção)

<sup>7</sup> LTE (Limite de Tolerância de Especificação); LTP (Limite de Tolerância de Produção)

<sup>8</sup> Tempo fora dos limites de controle (nº de vezes que o solver foi iniciado) Desvio Total Acumulado para todas as peneiras

variação em seus insumos de entrada, quando comparado com o sistema com uma variação menor, acionando o Solver mais vezes e gerando instabilidades no sistema.

Em todas as situações testadas a estratégia de otimização com o solver apresentou uma redução tanto no tempo fora de controle quanto no desvio total acumulado. Este resultado demonstra a validade e efetividade do sistema proposto.

## 5. CONSIDERAÇÕES FINAIS SOBRE A DISSERTAÇÃO

### 5.1. CONCLUSÕES

Este trabalho descreve um sistema de controle online em tempo real da dosagem dos agregados de entrada para um processo de usina de asfalto. O *software* livre Ururau foi utilizado para simular o processo, trocar dados e manipular planilhas em Excel, acionando o complemento “Solver” para otimizar a mistura em tempo real. Uma das principais dificuldades nos processos de dosagem é a variabilidade em seus insumos de entrada e a capacidade de identificar e corrigir rapidamente este desvio. Assim como relatou Kabadurmus *et al.* (2010) e Zhang *et al.* (2014) o sistema apresentado demonstrou ser eficaz nesta correção durante a produção, garantindo uma padronização e um aumento de qualidade no produto final.

As aplicações desenvolvidas no Ururau de troca de dados e execução de macros em planilhas do Excel demonstraram bastante agilidade durante a execução da simulação em tempo real. Não foram observados travamentos nem prejuízos na execução da simulação do processo quando esta aplicabilidade foi integrada juntamente com a simulação e a troca de dados com o CLP.

A aplicação para o processo de usina de asfalto se mostrou eficiente, porém este princípio pode ser aplicado a outros processos de dosagem de insumos. Conhecendo as variáveis, curvas e restrições dos processos, este sistema se mostra flexível e seguro, pois permite testes de lógica, comissionamentos e ajustes no sistema de controle, sem gerar prejuízos no sistema real, além do código do

*software* Ururau ser aberto, possibilitando o usuário adaptar o *software* à sua preferência e realizar uma infinidade de testes. O objetivo deste novo software de simulação é a disseminação, utilização e compreensão da Simulação de Eventos Discretos no Brasil, permitindo os usuários terem contato com este *software* específico de simulação, desenvolvendo suas capacidades e tendo a possibilidade de conhecer a concepção interna da estrutura do *software*.

## 5.2. LIMITAÇÕES DO TRABALHO

A pesquisa se limita a utilizar cenários hipotéticos de uma usina de asfalto, mesmo os dados sendo referentes a sistemas reais, não foi utilizado um sistema real para se testar a otimização com o Solver proposta.

Não foram realizados testes exaustivos com o *software* Ururau. Foi realizado apenas uma rodada de simulação para cada cenário testado. O *software* Ururau ainda é uma ferramenta em desenvolvimento, sendo assim a proposta da pesquisa é verificar o comportamento do *software* neste tipo de aplicação. Modelos mais complexos podem ser bastante trabalhosos ou inviáveis de se desenvolver neste *software*.

## 5.3. SUGESTÃO PARA TRABALHOS FUTUROS

Como exposto anteriormente, foi utilizado um sistema hipotético, portanto em trabalhos futuros pode-se aplicar a otimização proposta em sistemas reais. Poderia se comparar os resultados obtidos pelo *software* Ururau com outros *softwares* do mercado.

Os testes poderiam ser realizados com outros sistemas de dosagem para verificar se os resultados também seriam positivos envolvendo as particularidades de outro processo.

Uma sugestão ainda seria realizar diversas rodadas de simulação e testar outros critérios de controle e verificar o comportamento do processo.

### 5.3. PRODUÇÕES ORIGINADAS NESTE TRABALHO

ARTIGO	SITUAÇÃO
FRAGA, G.R.; RANGEL, J.J.A.; PEIXOTO, T.A. Simulation Optimization in Dosing Process Control System in Real Time in a Free Open Source Software	A ser submetido no Journal of Simulation

**Quadro 4.** Produções Originadas Neste Trabalho.  
Fonte: Elaborado pelo Autor (2017)

## 6. REFERÊNCIAS BIBLIOGRÁFICAS

ALVAREZ, A.M.; CENTENO, M.A. Enhancing simulation models for emergency rooms using VBA. In: WINTER SIMULATION CONFERENCE, 1999, 05-08 dez; Phoenix, AZ, USA. **Proceedings...** New York: IEEE, 2000. p. 1685-1693. Disponível em: <[https://www.researchgate.net/publication/2471604\\_Enhancing\\_Simulation\\_Models\\_For\\_Emergency\\_Rooms\\_Using\\_Vba](https://www.researchgate.net/publication/2471604_Enhancing_Simulation_Models_For_Emergency_Rooms_Using_Vba)>. Acesso em: 12 jan. 2017.

BANKS, J.; CARSON, S. J.; NELSON, B. L.; NICOL, D. **Discrete-event system simulation**. 2 ed. New York: Prentice Hall, 2010.

BROOKS, B.; DAVIDSON A.; MC GREGOR, I. The evolving relationship between simulation and emulation: faster than real-time controls testing. In: WINTER SIMULATION CONFERENCE, 2014, 07-10 dez; Savannah, GA. **Proceedings...** New York: IEEE, 2015. p. 978-981. Disponível em: <<https://pdfs.semanticscholar.org/1799/125db9c5d93def993617e7f20be8c2b56cbd.pdf>>. Acesso em: 21 nov 2016.

CARDOSO, L.D.D.; et al. Discrete event simulation for teaching in control systems. In: WINTER SIMULATION CONFERENCE, 2015, 06-09 dez, Huntington, CA. **Proceedings ....** New York: IEEE, 2016. p. 3608-3617. Disponível em: <<http://informs-sim.org/wsc14papers/includes/files/321.pdf>>. Acesso em: 12 dez 2016.

COSTA, G.O.; SANTOS, E.A.P.; BUSETTI, M.A. Cost based implementation of modular supervisory control theory. **IFAC**, v. 38, n.1, p.78-83., 2005. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1474667016374079>>. Acesso em: 12 dez 2016.

CHRISTENSEN, D. A mix design manual for Hot Mix Asphalt: Draft Report. In: NATIONAL COOPERATIVE HIGHWAY RESEARCH PROGRAM (NCHRP). **2010 Summary of Progress**. Washington, D.C: Transportation Research Board, 2010. Disponível em: <<http://onlinepubs.trb.org/onlinepubs/nchrp/nchrpannual2010.pdf>>. Acesso em: 13 dez 2017.

DAGKAKIS, G.; HEAVEY, C. A review of open source discrete event simulation software for operations research. **Journal of Simulation**, p. 1–14, 2015. Disponível em: < <https://link.springer.com/article/10.1057/jos.2015.9> >. Acesso em: 13 dez 2017.

QUEIROZ, M.H. de; CURY, J.E.R. Synthesis and implementation of local modular supervisory control for a manufacturing cell. In: INTERNATIONAL WORKSHOP ON DISCRETE EVENT SYSTEMS, 6, 2002, 2-4 oct, Zaragoza, Spain. **Proceedings...** Zaragoza, Spain: Universidad de Zaragoza, 2003. p. 377-382, 2002. Disponível em: <[http://home.deib.polimi.it/ferrarin/supercontrol/A\\_Queroz-Cury-WODES2002.pdf](http://home.deib.polimi.it/ferrarin/supercontrol/A_Queroz-Cury-WODES2002.pdf) >. Acesso em: 21 jan. 2017.

FABIAN, M.; HELLGREN, A. PLC-based implementation of supervisory control for discrete event systems. In: IEEE CONFERENCE ON DECISION AND CONTROL, 1998, 18 dec, Tampa, FL, USA. **Proceedings...** New York: IEEE, 1999. p. 3305-3310. Disponível em: < <http://ieeexplore.ieee.org/document/758209/>>. Acesso em: 21 jan. 2017.

FREITAS FILHO, P. J. **Introdução à modelagem e simulação de sistemas com aplicações Arena**. 2. ed. São Paulo: Visual Books, 2008.

FU, M. C. et al. Integrating optimization and simulation: Research and practice. In: Winter Simulation Conference, 32, 2000, 10-13 dez; Orlando FL,USA. **Proceedings....** New York: IEEE, 2001. p. 610-616. Disponível em: <<http://dl.acm.org/citation.cfm?id=510468>>. Acesso em: 21 jan. 2017.

HENRIKSEN, S.J; et al. **Application of a genetic algorithm in crane movement scheduling**. Sidney, Austrália: University of Newcastle, 1999. Disponível em: <<http://sigpromu.org/reports/Document818.pdf>>. Acesso em: 12 jan 2017.

IMOLE, O. I.; et al. Experiments and discrete element simulation of the dosing of cohesive powders in a simplified geometry. **Powder Technology**, v 287, p 108-120, 2016. Disponível em: <[https://www2.msm.ctw.utwente.nl/oimole/Papers/2014\\_Imole\\_Dosing.pdf](https://www2.msm.ctw.utwente.nl/oimole/Papers/2014_Imole_Dosing.pdf) >. Acesso em: 12 nov 2016.

JACOBS, P.H.M.; VERBRAECK, A.; RENGELINK, W. Emulation with DSOL. In: WINTER SIMULATION CONFERENCE, 2005, 22-24 dez, Orlando, FL, USA. **Proceedings** ... New York: IEEE, 2006. p. 1453-1462,. Disponível em: <<http://dl.acm.org/citation.cfm?id=510468>>. Acesso em: 21 jan. 2017.

JAHANGIRIAN, et al. Simulation in manufacturing and business: a review. **European Journal of Operational Research**, v. 203, n.1, p. 1-13, 2010. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0377221709004263>>. Acesso em: 20 jan 2017.

JOHNSTONE, M.; CREIGHTON, D.; NAHAVANDI, S. Enabling industrial scale simulation / emulation models. In: WINTER SIMULATION CONFERENCE., 2007, 09-12 dez; Washington, DC, USA. **Proceedings...** New York: IEEE, 2008. p. 1028-1034. Disponível em: <<http://www.informs-sim.org/wsc07papers/120.pdf>>. Acesso em: 21 jan. 2017.

KABADURMUS, O.; et al. A simulation methodology for online process control of hot mix asphalt (HMA) production. In: WINTER SIMULATION CONFERENCE, 2010, 5-8 dec; Baltimore, Maryland, USA. **Proceedings...** New York: IEEE, 2011. p. 1522-1533. Disponível em: <<http://www.informs-sim.org/wsc10papers/140.pdf>>. Acesso em: 13 set 2016.

KELTON, W.D.; SADOWSKI, R.P.; SWETS, N.B. **Simulation with Arena**. 5 ed. Boston-USA: McGraw-Hill, 2010.

KWAN, A. K. H.; MORA, C. F.; CHAN, H. C. Particle shape analysis of coarse aggregate using Digital Image Processing. **Cement and Concrete Research**, p.1403-1410, 1999. Disponível em: <<http://hub.hku.hk/handle/10722/71774> >. Acesso em: 12 out 2016.

MAHDAVI, I.; SHIRAZI, B.; SOLIMANPUR, M. Development of a simulation-based decision support system for controlling stochastic flexible job shop manufacturing systems. **Simulation Modelling Practice and Theory**, p.768-786, 2010. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1569190X10000274>>. Acesso em: 12 out 2016.

MONTEVECHI, J.A.B. et al. Conceptual modeling in simulation projects by mean adapted IDEF: an application in a Brazilian tech company. In: WINTER SIMULATION CONFERENCE, 2010, 5-8 dec; Baltimore, Maryland, USA. **Proceedings...** New York: IEEE, 2011. p. 1624-1635, 2010. Disponível em: <<https://pdfs.semanticscholar.org/5a3a/c200fbb7f8a128d27f42f49fedb84e4b32da.pdf> >. Acesso em: 12 dez 2016.

OESTERREICH, T. D.; TEUTEBERG, F. Understanding the implications of digitisation and automation in the context of Industry 4.0: a triangulation approach and elements of a research agenda for the construction industry. **Computers in Industry**, v. 83, p. 121-139, dec; 2016. Disponível em: <<http://dl.acm.org/citation.cfm?id=3083320>>. Acesso em: 12 jan 2017.

PARK, S.C; et al. PLC Studio: Simulation based PLC code verification. In: WINTER SIMULATION CONFERENCE, 2008, 07-10 dec; Miami, FL, USA. **Proceedings...** New York: IEEE, 2009. p. 222-228. Disponível em: <<http://simulation.su/uploads/files/default/2008-park-park-wang-kwak-yeo.pdf>>. Acesso em: 19 jan 2017.

PEIXOTO, T. A.; et al. Ururau: a free and open-source discrete event simulation software. **Journal of Simulation**, 2016. Disponível em: < doi:10.1057/s41273-016-0038-5>. Acesso em: 20 jan 2017.

RAMADGE, P. J. C.; WONHAM, W. M. The control of discrete event systems. **Proceedings of the IEEE**, v.77, n.1, p. 81-98, 1989. Disponível em: <<http://www.labri.fr/perso/anca/Games/RamadgeWonham89.pdf> >. Acesso em: 13 jan 2017.

RANGEL, J. J. de A.; et al. Simulação a eventos discretos para treinamento em sistemas de controle. **Revista Eletrônica Pesquisa Operacional para o Desenvolvimento**, Rio de Janeiro, v.4, n.1, p. 97-111, 2012. Disponível em: <<http://www.podesenvolvimento.org.br/inicio/index.php?journal=podesenvolvimento&page=article&op=view&path%5B%5D=146>>. Acesso em: 12 jan 2017.

RIERA, B.; et al. A non-intrusive method to make safe existing PLC program. **IFAC**, p. 320-325, 2015. Disponível em: <[https://www.researchgate.net/publication/283989857\\_A\\_non-intrusive\\_method\\_to\\_make\\_safe\\_existing\\_PLC\\_Program](https://www.researchgate.net/publication/283989857_A_non-intrusive_method_to_make_safe_existing_PLC_Program) >. Acesso em: 12 jan 2017.

ROSSETTI, M.D. Java Simulation Library (JSL): an open-source object-oriented library for discrete-event simulation in Java. **International Journal Simulation and Process Modelling**, v. 4, n. 1, p. 69-87, 2008. Disponível em: <<https://pdfs.semanticscholar.org/2d56/0eb6a338588722a5ec8c6dacab6d6e40a317.pdf> >. Acesso em: 12 dez 2016.

SARGENT, R. G. Verification and validation of simulation models. In: WINTER SIMULATION CONFERENCE, 2008, 11-14 dec; Phoenix, AZ, USA. **Proceedings...** New York: IEEE, 2009. p.183-198 Disponível em: <<http://www.informs-sim.org/wsc11papers/016.pdf> >. Acesso em: 12 dez 2016.

SCHLUDERMANN, H.; KIRCHMAIR, T.; VORDERWINKLER, M. Soft-Commissioning: Hardware-in-the-loop-based verification of controller software. In: WINTER SIMULATION CONFERENCE, 2000, 11-14 dec; Orlando, FL, USA. **Proceedings...** New York: IEEE, 2001. p. 893-899. Disponível em: <<http://ieeexplore.ieee.org/document/899889/>>. Acesso em: 12 dez 2016.

SCHUH, G.; REUTER, C.; HAUPTVOGEL, A.; DÖLLE, C. Hypotheses for a Theory of Production in the Context of Industrie 4.0. **Advances in Production Technology**, 11-23 p; 2015. Disponível em: <[http://www.springer.com/cda/content/document/cda\\_downloaddocument/9783319123035-c2.pdf?SGWID=0-0-45-1487582-p177033750](http://www.springer.com/cda/content/document/cda_downloaddocument/9783319123035-c2.pdf?SGWID=0-0-45-1487582-p177033750)>. Acesso em: 23 jan 2017.

SEPPANEN, M.S. Developing industrial strength simulation models using Visual Basic for Applications (VBA). In: WINTER SIMULATION CONFERENCE, 2000, 11-14 dec; Orlando, FL, USA. **Proceedings...** New York: IEEE, 2001, p. 77-82. Disponível em: <<https://pdfs.semanticscholar.org/eb6f/97a12272488d11bc2286148d7f409c22c4b4.pdf>>. Acesso em: 12 dez 2016.

SILVA, F. F. et al. Software Ururau: produto tecnológico. In: ENCONTRO DE PESQUISA E PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO, 2017, 15-17 fev; Curitiba-PR. **Anais...** Rio de Janeiro: ANPEPRO, 2017. Disponível em: <<http://anpepro.org.br/wp-content/uploads/2016/06/EPPGEP-2017-Abstract-proceedings.pdf>>. Acesso em: 03 abr. 2017.

SMITH, J.S.; CHO Y. Offline commissioning of a plc-based control system using arena. In: WINTER SIMULATION CONFERENCE, 2008, 11-14 dec; Phoenix, AZ, USA. **Proceedings...** New York: IEEE, 2009.p. 1802-1810. Disponível em: <<https://pdfs.semanticscholar.org/ba5e/72e87a0757a1957a9e78346523a831be2afe.pdf>>. Acesso em: 03 abr. 2017.

WANG, X.; TAKAKUWA, S. Module-based modeling of production-distribution systems considering shipment consolidation. In: WINTER SIMULATION CONFERENCE, 2008, 03-06 dec; Monterey, California, USA. **Proceedings...** New York: IEEE, 2007. p. 1477-1484. Disponível em: <<https://pdfs.semanticscholar.org/8dbd/42be4070c095223c1c9b81e1270c91fe13e5.pdf>>. Acesso em: 21 jan 2017.

WILSON, J.; et al. A simple energy usage toolkit from manufacturing simulation data. **Journal of Cleaner Production**, p. 266-276, 2016. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0959652615017412>>. Acesso em: 21 jan 2017.

XU, J.; et al. Simulation optimization in the era of Industrial 4.0 and the Industrial Internet. **International Journal Simulation and Process Modelling**, v. 10, n. 4, p. 310-320, 2016. Disponível em: < <https://link.springer.com/article/10.1057/s41273-016-0037-6>>. Acesso em: 21 jan 2017.

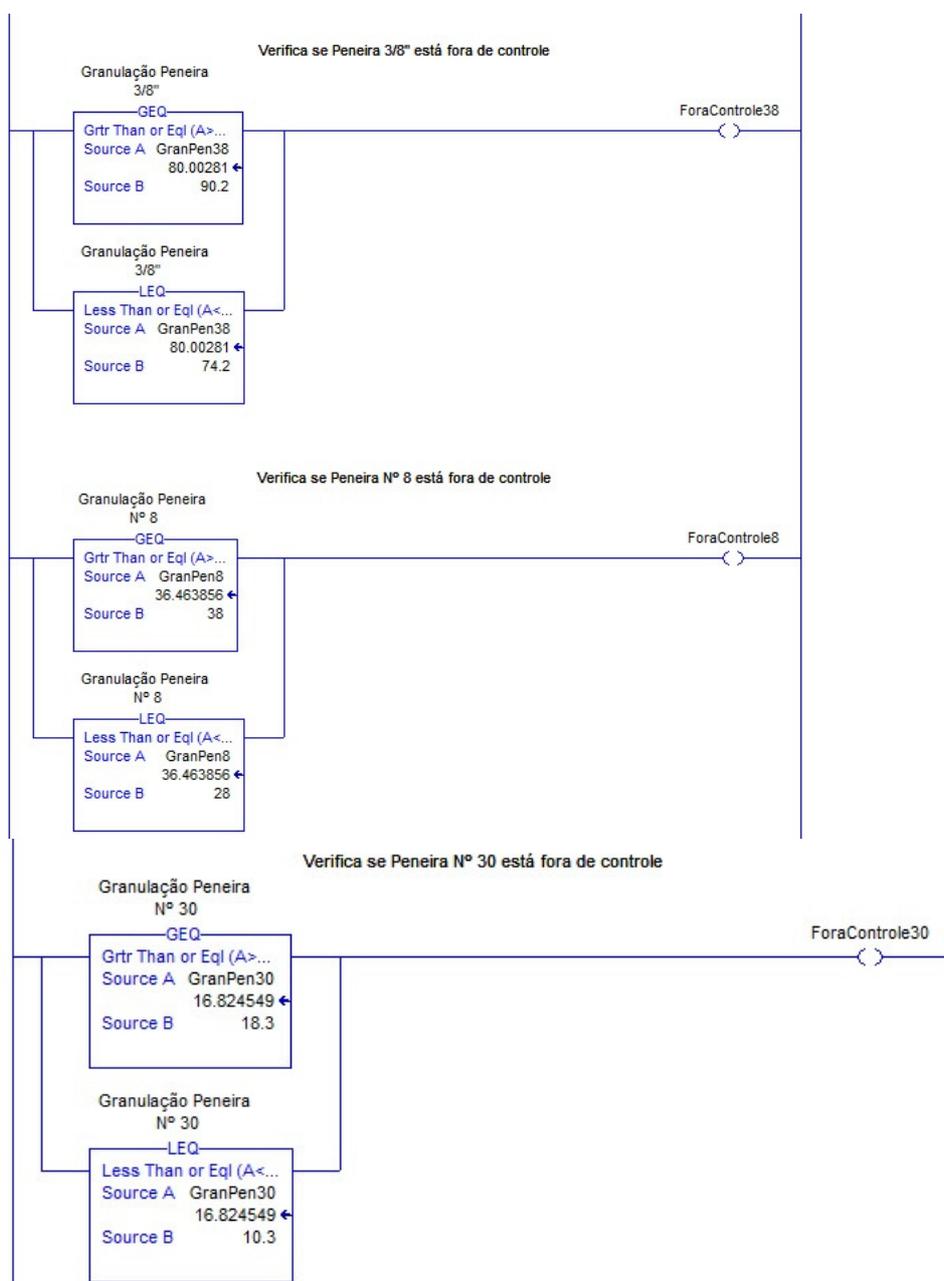
ZHANG, M.; HEITZMAN, M.; SMITH, A. E. Improving hot mix asphalt production using computer simulation and real time optimization. **Journal of Computing in Civil Engineering**, v. 28, n.3, may, 2014. Disponível em: <<http://ascelibrary.org/doi/full/10.1061/%28ASCE%29CP.1943-5487.0000302#sthash.zaARNqt5.dpuf>>. Acesso em: 21 jan 2017.

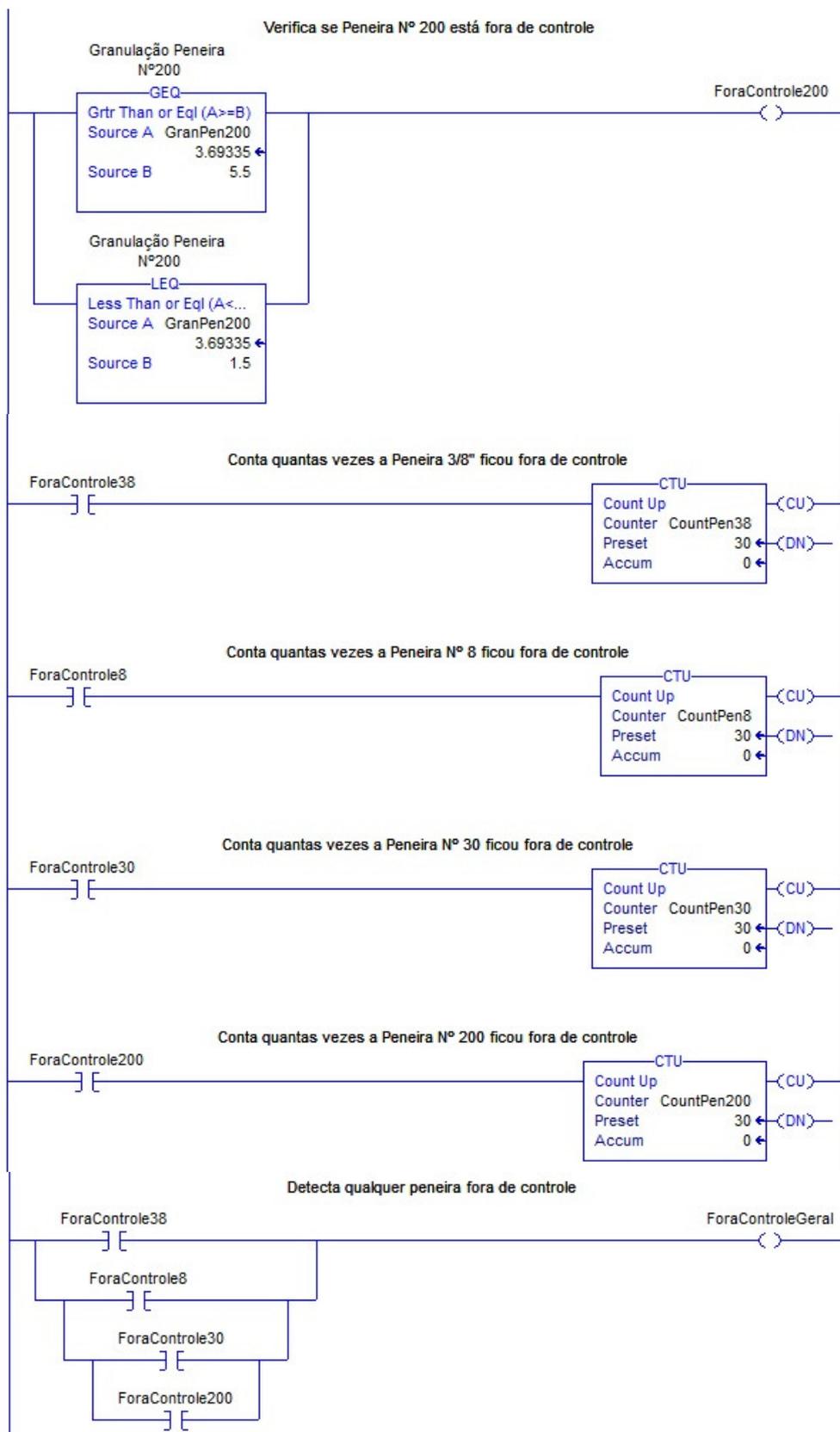
## APÊNDICE A: PARÂMETROS DA SIMULAÇÃO

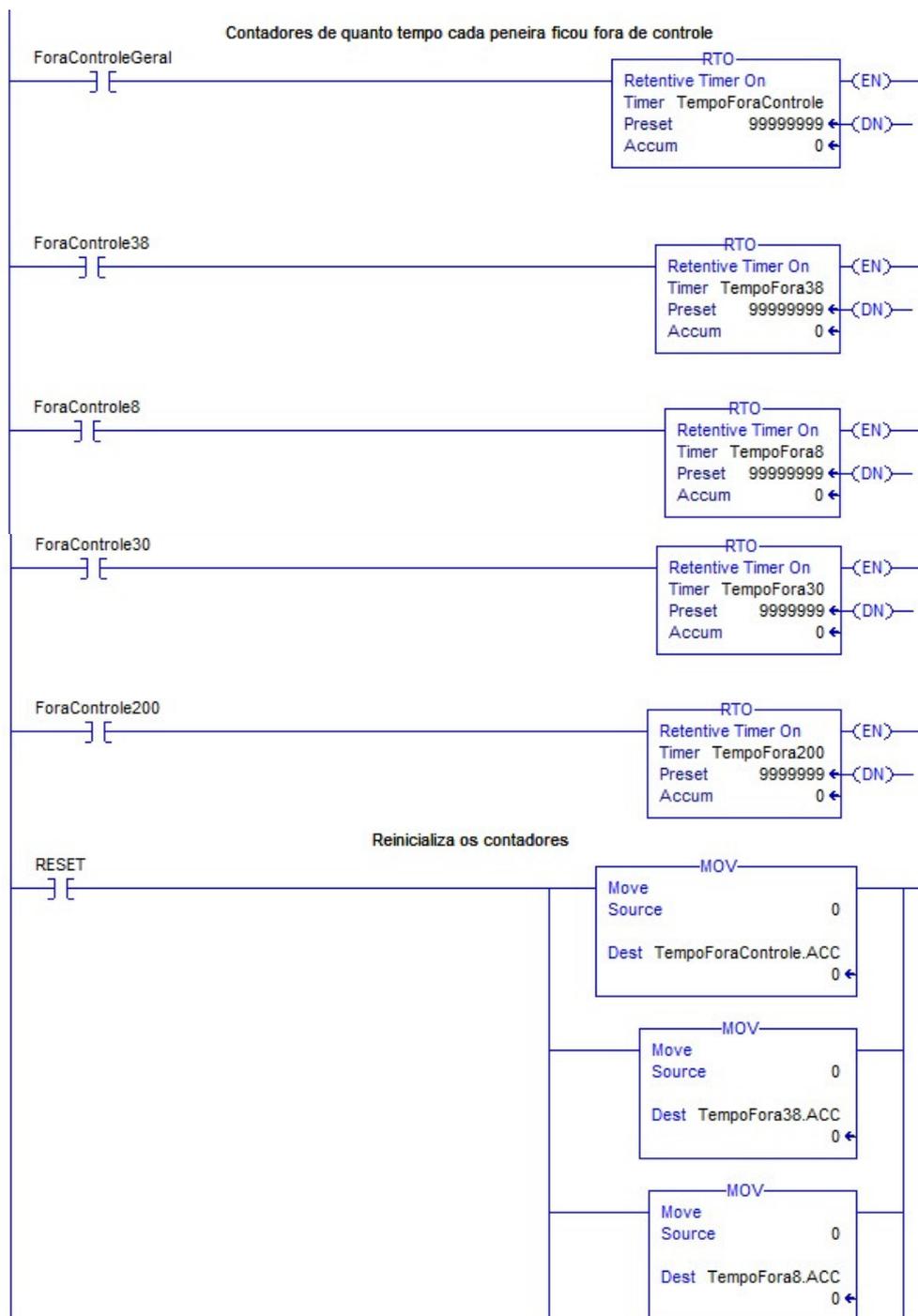
Nome	Função	Parâmetros
E1	Create	Constant (1)min
L1	Hold	if Release == 0
CT1	OR	(TNOW <=120)    ((TNOW> 240) && (TNOW<=360))    (TNOW>480)
A1	Assign (Variable)	Release == 1
A2	Assign (Variable)	Function1 == 1.33
A3	Assign (Variable)	Function2 == 0.67
A4	Assign (Variable)	Function1 == 0.67
A5	Assign (Variable)	Function2 == 1.33
CT2	OR	100% (Union)
A6	Assign (Variable)	Var1Sieve38 == Function2*NORM(52,3)
A7	Assign (Variable)	Var1Sieve8 == Function2*NORM(14,2)
A8	Assign (Variable)	Var2Sieve8 == Function1*NORM(96,1)
A9	Assign (Variable)	Var3Sieve8 == Function2*NORM(86,2)
A10	Assign (Variable)	Var4Sieve8 == Function1*NORM(7,1)
A11	Assign (Variable)	Var5Sieve8 == Function2*NORM(52,3)
A12	Assign (Variable)	Var1Sieve30 == Function2*NORM(7.9,1)
A13	Assign (Variable)	Var2Sieve30 == Function1*NORM(51,4)
A14	Assign (Variable)	Var3Sieve30 == Function2*NORM(41,3)
A15	Assign (Variable)	Var4Sieve30 == Function1*NORM(1,0.5)
A16	Assign (Variable)	Var5Sieve30 == Function2*NORM(9,2)
A17	Assign (Variable)	Var1Sieve200 == Function2*NORM(3.7,1)
A18	Assign (Variable)	Var2Sieve200 == Function1*NORM(14,2)
A19	Assign (Variable)	Var3Sieve200 == Function2*NORM(1.4,0.5)
A20	Assign (Variable)	Var4Sieve200 == Function1*NORM(0.1,0.05)
A21	Assign (Variable)	Var5Sieve200 == Function2*NORM(0.7,0.01)
RW1	Read/Write in Excel	Write all sieves functions in Excel
RM1	Run Macro in Excel	VBA "Application.CalculateFull"
RW2	Read/Write in Excel	Read Sieves Gradation, Deviations, # Sieve Out Specifications, % Bin Openings
A22	Assign (Variable)	AcumDeviation == AcumDeviation + TotalDeviation

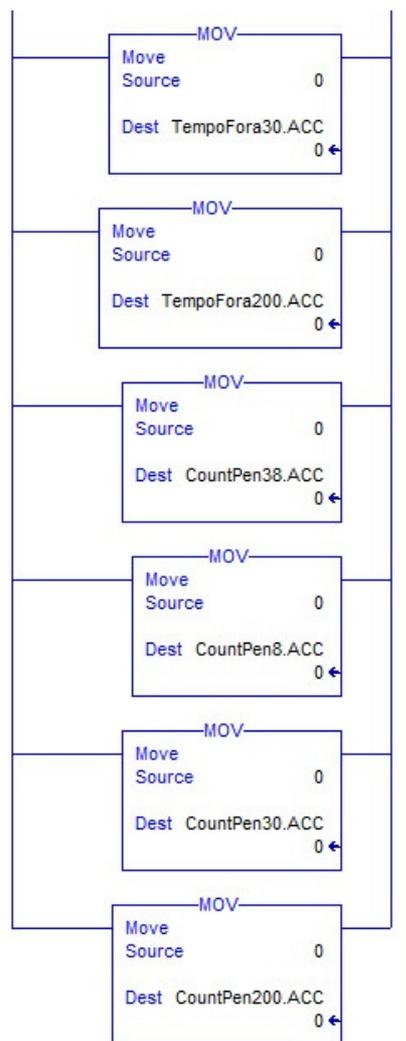
C1	Communication (Real Tag)	BinOpen1
C2	Communication (Real Tag)	BinOpen2
C3	Communication (Real Tag)	BinOpen3
C4	Communication (Real Tag)	BinOpen4
C5	Communication (Real Tag)	BinOpen5
C6	Communication (Real Tag)	AcumDeviation
C7	Communication (Real Tag)	Bin1Sieve38
C8	Communication (Real Tag)	Bin2Sieve38
C9	Communication (Real Tag)	Bin3Sieve38
C10	Communication (Real Tag)	Bin4Sieve38
C11	Communication (Real Tag)	Bin5Sieve38
C12	Communication (Real Tag)	GradationSieve38
C13	Communication (Real Tag)	DeviationSieve38
C14	Communication (Real Tag)	Bin1Sieve8
C15	Communication (Real Tag)	Bin2Sieve8
C16	Communication (Real Tag)	Bin3Sieve8
C17	Communication (Real Tag)	Bin4Sieve8
C18	Communication (Real Tag)	Bin5Sieve8
C19	Communication (Real Tag)	GradationSieve8
C20	Communication (Real Tag)	DeviationSieve8
C21	Communication (Real Tag)	Bin1Sieve30
C22	Communication (Real Tag)	Bin2Sieve30
C23	Communication (Real Tag)	Bin3Sieve30
C24	Communication (Real Tag)	Bin4Sieve30
C25	Communication (Real Tag)	Bin5Sieve30
C26	Communication (Real Tag)	GradationSieve30
C27	Communication (Real Tag)	DeviationSieve30
C28	Communication (Real Tag)	Bin1Sieve200
C29	Communication (Real Tag)	Bin2Sieve200
C30	Communication (Real Tag)	Bin3Sieve200
C31	Communication (Real Tag)	Bin4Sieve200
C32	Communication (Real Tag)	Bin5Sieve200
C33	Communication (Real Tag)	GradationSieve200
C34	Communication (Real Tag)	DeviationSieve200
C35	Communication (Real Tag)	TotalDeviation
OR2	OR	# Sieve Out Specifications >= 1
RM2	Run Macro in Excel	VBA "SolverSolve(True)"
RW3	Read/Write in Excel	Read Sieves Gradation, Deviations, # Sieve Out Specifications, % Bin Openings
R1	Counter	CounterSolver == CounterSolver + 1
R2	Counter	CounterOK == CounterOK + 1
A23	Assign (Variable)	Release == 0

## APÊNDICE B: PROGRAMAÇÃO EM LADDER









## APÊNDICE C: OTIMIZAÇÃO COM SOLVER DO EXCEL

	A	B	C	D	E	F
1			Peneiras			
2	Critério de Agregados	Silos	3/8"	Número 8	Número 30	Número 200
3	Granulação Instântanea (Simulação)	1	35,8791706	7,952809653	3,883412578	2,491078152
4		2	100	126,9960228	63,27996509	19,75166341
5		3	100	58,40657773	25,47512382	0,063485859
6		4	100	11,30173421	1,113738182	0,164828614
7		5	100	36,16214134	5,470476999	0,460842074
8						
9	<b>Abertura dos Silos</b>		Limite percentual dos silos			
10	Silos	Proporção Inicial	Mínimo	Máximo		
11	1	37	17	57		
12	2	14	5	23		
13	3	7	2	12		
14	4	30	10	50		
15	5	12	5	23		
16	SOMATÓRIO	100				
17						
18	Meta de porcentagem das peneiras (PM)		82,2	33	14,3	3,5
19	Limites de tolerância da especificação		8	5	4	2
20	Limite Inferior		74,2	28	10,3	1,5
21	Limite Superior		90,2	38	18,3	5,5
22						
23		Silos	Somatório Silo x Granulação Peneiras			
24		1	13,27529312	2,942539572	1,436862654	0,921698916
25		2	14	17,77944319	8,859195113	2,765232878
26		3	7	4,088460441	1,783258668	0,00444401
27		4	30	3,390520263	0,334121454	0,049448584
28		5	12	4,339456961	0,65645724	0,055301049
29	SOMATORIO		76,27529312	32,54042043	13,06989513	3,796125437
30						
31	Estrutura Função Objetivo		0,74	0,09	0,31	0,15
32	Minimizar Somatório Função Objetivo		1,28809321			
33						
34	Verifica se está dentro dos limites	Se maior	0	0	0	0
35		Se menor	0	0	0	0
36					TOTAL FORA	0

A simulação envia as informações de granulação das peneiras para as células "C3:F7" em destaque na cor azul.

As células variáveis “B11:B15” que são a solução da otimização do Solver estão em destaque na cor amarela e correspondem aos valores de porcentagem de abertura dos silos.

As células “C24:F28” são a multiplicação das granulações enviadas pela simulação (células “C3:F7”) com os valores de percentual de abertura dos silos (células “B11:B15”).

A linha “Estrutura Função Objetivo” tem como objetivo estruturar a fórmula da Função Objetivo que será otimizada (esta função é a número “1” e está localizada na página 63). Para exemplificar a célula “C31” contém a seguinte fórmula na sua estrutura:  $=\text{ABS}(C18-C29)/(((C21)-(C20))/2)$ .

A linha “Minimizar Somatório Função Objetivo” apenas soma os valores da linha “Estrutura Função Objetivo” e é a função objetivo configurada no Solver para ser minimizada e está em destaque na planilha na cor verde.

A estrutura que verifica se o sistema possui alguma peneira fora de controle apenas verifica se os valores de somatório das granulações das peneiras contidos nas células “C29:F29” está dentro dos limites inferior e superior contidos nas células “C20:F21”. A cada rodada de simulação, o Ururau verifica se o valor de peneiras fora de controle é maior ou igual a um para que seja iniciado o processo de otimização pelo Solver.