



Universidade Federal da Bahia  
Instituto de Computação / Escola Politécnica

Programa de Pós-Graduação em Mecatrônica

**SAMOM: UMA ARQUITETURA PARA OPERAÇÃO DE SISTEMAS DE  
MANUFATURA COM HABILIDADES INTELIGENTES E ADAPTATIVAS**

Jhaidan Ribeiro Cruz

DISSERTAÇÃO DE MESTRADO

Salvador  
12 de agosto de 2024

JHAIDAN RIBEIRO CRUZ

**SAMOM: UMA ARQUITETURA PARA OPERAÇÃO DE SISTEMAS DE  
MANUFATURA COM HABILIDADES INTELIGENTES E ADAPTATIVAS**

Dissertação apresentada ao Programa de Pós-Graduação em Mecatrônica, Escola Politécnica e Instituto de Computação, Universidade Federal da Bahia, como requisito para obtenção do título de Mestre em Mecatrônica.

Orientador: Prof. Herman Augusto Lepikson, Dr. Eng.

Salvador  
12 de agosto de 2024

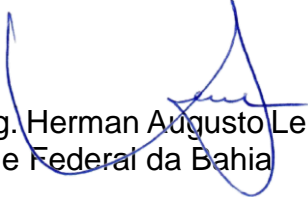
**Jhaidan Ribeiro Cruz**


**SAMOM: Uma Arquitetura para Operação de  
Sistemas de Manufatura com Habilidades  
Inteligentes e Adaptativas**


Dissertação apresentada como requisito para a obtenção do título de Mestre em Mecatrônica, Escola Politécnica e Instituto de Computação, da Universidade Federal da Bahia.

Salvador, 12 de agosto de 2024.


Banca examinadora

  
Prof. Dr.Eng. Herman Augusto Lepikson – Orientador  
Universidade Federal da Bahia

  
Prof. Dr. Ângelo Marcio Oliveiro Sant Anna  
Universidade Federal da Bahia

Documento assinado digitalmente  
 CARLOS ALBERTO TOSTA MACHADO  
Data: 13/08/2024 16:10:27-0300  
Verifique em <https://validar.it.gov.br>

Prof. Dr. Carlos Alberto Tosta Machado  
Universidade Federal do Recôncavo da Bahia.

DocuSigned by:  
  
4C1ACA2A8BB044B...

Prof<sup>a</sup>. Dr<sup>a</sup>. Thamiles Rodrigues de Melo  
Centro Universitário Senai CIMATEC

Ficha catalográfica elaborada pela Biblioteca Bernadete  
Sinay Neves, Escola Politécnica – UFBA.

---

C957 Cruz, Jhaidan Ribeiro.

Samom: uma arquitetura para operação de sistemas de  
manufatura com habilidades inteligentes e adaptativas / Jhaidan  
Ribeiro Cruz. – Salvador, 2024.

124f. : il. color.

Orientador: Prof. Dr. Herman Augusto Lepikson.

Dissertação (mestrado) – Programa de Pós-Graduação em  
Mecatrônica, Escola Politécnica e Instituto de Computação,  
Universidade Federal da Bahia, 2024.

1. Controle de processos. 2. Manufaturas. 3. Indústria 4.0. 4.  
Sistemas multiagentes. I. Lepikson, Herman Augusto. II.  
Universidade Federal da Bahia. III. Título.

---

CDD: 670

*Não pense que a história dos outros é mais interessante do que a sua, pois ninguém sabe a real profundidade das situações que o outro passa.*

*Ame a sua história, ela é linda, ela é sua!*

*(Jhaidan Ribeiro Cruz)*

# AGRADECIMENTOS

A Deus, por me trazer até aqui, me capacitando para superar as dificuldades, me dando forças para suportar as cargas emocionais e físicas, e por me proporcionar o privilégio de viver essa experiência, bem como outras, imerecidamente.

À minha mãe, Tainara Souza Ribeiro Cruz, por todo esforço, luta e renúncia, por tudo que caracteriza o seu amor puro por mim.

Ao meu pai, Juscelino dos Santos Cruz, por ser exemplo de caráter e humildade, pela educação, pelos sacrifícios que faz para me ajudar, por todo amor oferecido.

À minha irmã Thaiana Ribeiro Cruz por toda ajuda e apoio, pelos bons momentos, pela preocupação, carinho e amor.

À minha amada noiva Isadora Nadinne Ribeiro da Palma, pelo amor, motivação, carinho, preocupação, incentivo, por compartilhar alegrias e tristezas, por viver comigo essa jornada rumo a nossos sonhos. Ao infinito e amém!

Ao meu orientador, Prof. Herman Augusto Lepikson, pela grande disponibilidade de compartilhar o seu conhecimento e sua experiência com humildade e empatia, por ser mais que um profissional excelente: um exemplo de dedicação e alegria pelo que faz.

Aos amigos Elisabete Guedes Conceição e Paulo Henrique Farias de Carvalho Filho, pelos bons momentos, companheirismo, empatia, por dividir as dificuldades do mestrado e por tornarem tudo mais leve e feliz.

Ao meu amigo George Gonçalves dos Santos, pelo companheirismo nos diversos momentos, pelos conselhos, pela ajuda, pelas cobranças.

Ao SENAI CIMATEC e aos amigos da Automação Industrial, em especial os que compartilharam momentos comigo na PMA, pela disponibilidade em ajudar, pela oportunidade e por ceder o campo para a pesquisa.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, à Universidade Federal da Bahia e ao Programa de Pós-Graduação em Mecatrônica, pelo fomento e pela oportunidade de realizar a pesquisa.

A todos os meus familiares pela preocupação, apoio e incentivo.

A todos os meus amigos próximos, pela companhia e apoio.

A todos que contribuíram direta e indiretamente com minha jornada no mestrado.

## RESUMO

Os sistemas de manufatura comumente necessitam de adequada gestão e controle (G&C) para tomadas de decisão e conseqüente atendimento das demandas de produção. Os sistemas anteriores à Indústria 4.0 (tradicionais), são limitados quanto à flexibilidade necessária para se adequarem às demandas emergentes, como a diversidade de processos e a personalização de produtos. Nestes sistemas, as características típicas de uma automação dedicada e focada em ilhas, pouco favorece a uma integração adequada. Em conseqüência, as atividades de G&C despendem elevadas quantidades de recursos. A Indústria 4.0 (I4.0) traz conceitos e tecnologias, que oportunizam novos sistemas de manufatura (sistemas de manufatura 4.0) com maior flexibilidade e eficiência de G&C. O presente trabalho tem como objetivo propor uma arquitetura para gestão e controle integrados de sistemas de manufatura discretos e em batelada com características distribuídas e com habilidades preditivas e adaptativas. Para isso, a arquitetura se utiliza de tecnologias habilitadoras e conceitos da I4.0, com foco em Análise de Dados, IA, Integração de Sistemas e Arquitetura Orientada a Serviços (SOA – *Service-Oriented Architecture*). O modelo proposto se desdobra numa SOA, baseada em Sistemas Multiagentes (SMA) e Sistemas de Execução da Manufatura (MES – *Manufacturing Execution Systems*). Também descreve componentes, interações, conceitos e particularidades de implementação de tal arquitetura. O modelo proposto foi validado em uma planta piloto de manufatura 4.0, em que foi possível demonstrar que os dispositivos de tal sistema foram gerenciados e controlados, com o auxílio de representações digitais (agentes), de forma descentralizada, distribuída e heterárquica. Também foi possível demonstrar a realização, pelo SMA em questão, de previsões das disponibilidades dos dispositivos e de tomadas de decisão sobre o desempenho da produção. Conclui-se que o trabalho demonstra ser uma referência para implementação de arquiteturas compatíveis com a G&C de sistemas de manufatura 4.0.

**Palavras-chave:** Gestão e Controle de Processos, Sistemas de Execução da Manufatura, Arquitetura Orientada a Serviços, Adaptabilidade, Sistemas Multiagentes.



# ABSTRACT

## **SAMOM: AN ARCHITECTURE FOR OPERATING MANUFACTURING SYSTEMS WITH SMART AND ADAPTIVE ABILITIES**

Manufacturing systems commonly require adequate management and control (M&C) for decision-making and consequent meeting of production demands. Systems prior to Industry 4.0 (traditional) are limited in terms of the flexibility required to adapt to emerging demands, such as process diversity and product customization. In these systems, the typical characteristics of dedicated automation focused on islands do little to favor adequate integration. As a result, M&C activities require large amounts of resources. Industry 4.0 (I4.0) brings concepts and technologies that enable new manufacturing systems (4.0 manufacturing systems) with greater flexibility and M&C efficiency. This paper aims to propose an architecture for integrated management and control of discrete and batch manufacturing systems with distributed characteristics and predictive and adaptive capabilities. To this end, the architecture uses enabling technologies and concepts from I4.0, focusing on Data Analysis, AI, Systems Integration and Service-Oriented Architecture (SOA). The proposed model unfolds into a SOA, based on Multi-Agent Systems (MAS) and Manufacturing Execution Systems (MES). It also describes components, interactions, concepts and implementation particularities of such architecture. The proposed model was validated in a pilot plant of 4.0 manufacturing, where it was possible to demonstrate that the devices of such system were managed and controlled, with the help of digital representations (agents), in a decentralized, distributed and heterarchical way. It was also possible to demonstrate the realization, by the MAS in question, of predictions of the availability of the devices and of decision-making about the production performance. It is concluded that the work proves to be a reference for the implementation of architectures compatible with the G&C of manufacturing systems 4.0.

**Keywords:** Process Management and Control, Manufacturing Execution Systems, Service-Oriented Architecture, Adaptability, Multi-Agent Systems.

## LISTA DE FIGURAS

Figura 1.1 Fluxograma da metodologia do trabalho .....	10
Figura 1.2 Série de publicações por ano .....	13
Figura 1.3 Rede de palavras-chave .....	14
Figura 1.4 Quantidade de publicações por tipo .....	16
Figura 2.1 Mapa da fundamentação teórica e estado da arte .....	18
Figura 2.2 Comparativo de controle centralizado, descentralizado e distribuído e estruturas hierárquicas, semi-hierárquicas e heterárquicas .....	24
Figura 2.3 Estágios de maturidade tecnológica da I4.0.....	28
Figura 2.4 Integração em toda a empresa da arquitetura de agente de software .....	41
Figura 2.5 Arquitetura auto-organizada assistida por nuvem .....	42
Figura 2.6 Arquitetura de sistema de manufatura inteligente .....	44
Figura 2.7 Dispositivos de manufatura assistidos por computação de borda.....	46
Figura 2.8 Componentes do modelo conceitual IMES (MES Inteligente).....	48
Figura 3.1 Arquitetura Principal do SAMOM.....	52
Figura 3.2 Arquitetura dos DIM .....	53
Figura 3.3 Conjuntos de agentes e dispositivos físicos para um DIM genérico .....	54
Figura 3.4 Classes de agentes.....	58
Figura 4.1 Planta de Manufatura Avançada (PMA) do SENAI Cimatec .....	68
Figura 4.2 Peças fabricadas na PMA .....	69
Figura 4.3 Fluxo do processo da PMA .....	70
Figura 4.4 Pallet com peça na esteira .....	71
Figura 4.5 Arquitetura Convencional da PMA .....	72

Figura 4.6 Arquitetura Principal da instância na PMA .....	75
Figura 4.7 Arquitetura dos DIM da instância na PMA.....	76
Figura 4.8 Diagrama UML de agentes instanciados da PMA .....	77
Figura 4.9 Configuração de historiamto de variáveis (estado de estações) .....	80
Figura 4.10 Conjunto de dados historiados pelo TrakSYS (estado da estação de Saída).....	82
Figura 5.1 Operação dos agentes Atuadores (ambiente1.py) .....	84
Figura 5.2 Operação dos agentes Sensor Transporte e Gerenciador Planta (ambiente2.py) .....	85
Figura 5.3 Painel de indicadores da operação sob a Arquitetura Convencional .....	86
Figura 5.4 Captura da tela do AMES em operação junto ao SAMOM.....	86
Figura 5.5 Interface de programação do CLP6 (TIA Portal) .....	88

## LISTA DE QUADROS

Quadro 1.1 Trabalhos considerados como estado da arte.....	15
Quadro 2.1 Características dos sistemas de produção personalizados e dos tradicionais .....	26
Quadro 2.2 Síntese das principais indicações por trabalho .....	33
Quadro 2.3 Síntese das principais abordagens e lacunas encontradas, por trabalho .....	38
Quadro 5.1 Comparação do 1º critério: capacidade em ser distribuída, descentralizada e heterárquica .....	84
Quadro 5.2 Comparação do 2º critério: habilidades preditivas.....	85
Quadro 5.3 Comparação do 3º critério: habilidades adaptativas.....	87

## LISTA DE ACRÔNIMOS

AAS	<i>Asset Administration Shells</i>
ACF	agentes de chão de fábrica
AE	agente de estoque
AGV	<i>Automated Guided Vehicle</i> (Veículos Autônomoamente Guiados)
AM	agente de manutenção
AMES	<i>Advanced Manufacturing Execution System</i> (Sistema de Execução da Manufatura Avançada)
API	<i>Application Programming Interface</i> (Interfaces de Programação de Aplicações)
AR	Aprendizado Reforço (AR)
CLP	Controlador Lógico Programável
CMMS	<i>Computerized Maintenance Management Systems</i> (Sistema Informatizado de Gestão de Manutenção)
CNC	<i>Computer Numerical Control</i> (Controle Numérico Computadorizado)
CPPS	<i>Cyber-Physical Production System</i> (Sistema Ciber Físico de Produção)
CPS	<i>Cyber Physical Systems</i> (Sistemas Ciberfísicos)
DDS	<i>Data Distribution Service</i> (Serviço de Distribuição de Dados)
DIM	Dispositivos Inteligentes de Manufatura
EAM	<i>Enterprise Asset Management</i> (Gestão de Ativos Empresariais)
ERP	<i>Enterprise Resource Planning</i> (Planejamento de Recursos Empresariais)
FIPA-ACL	<i>Foundation for Intelligent Physical Agents – Agent Communication Language</i> (Linguagem de Comunicação de Agente da Fundação para Agentes Físicos Inteligentes)
G&C	gestão e controle
GD	Gêmeo Digital
HTTP	<i>Hypertext Transfer Protocol</i> (Protocolo de Transferência de Hipertexto)
I4.0	Indústria 4.0
IA	Inteligência Artificial
IEEE	<i>Institute of Electrical and Electronic Engineers</i> (Instituto de Engenheiros Eletricistas e Eletrônicos)
IHM	Interface Homem-Máquina
IoT	<i>Internet of Things</i> (Internet das Coisas)
ISA	<i>International Society of Automation</i> (ISA Sociedade Internacional de Automação)

JSON	<i>JavaScript Object Notation</i> (Notação de Objeto JavaScript)
KPI	<i>Key Performance Indicator</i> (Indicador Chave de Desempenho)
LIMS	<i>Laboratory Information Management System</i> (Sistema de Gerenciamento de Informações Laboratoriais)
LMAS	<i>Line-less Mobile Assembly</i> (Montagem Móvel Sem Linha)
M2M	<i>Machine to Machine</i> (Máquina a Máquina)
MES	<i>Manufacturing Execution System</i> (Sistema de Execução da Manufatura)
MESA	<i>Manufacturing Enterprise Solutions Association</i> (Associação de Soluções Empresariais de Manufatura)
ML	<i>Machine Learning</i> (Aprendizado de Máquina)
MOM	<i>Manufacturing Operations Management</i> (Gerenciamento das Operações de Manufatura)
MQTT	<i>Message Queuing Telemetry Transport</i> (Transporte de Telemetria de Enfileiramento de Mensagens)
MRP	<i>Material Requirement Planning</i> (Planejamento de Necessidade de Materiais)
OEE	<i>Overall Equipment Effectiveness</i> (Eficiência Global do Equipamento)
OPC UA	<i>Open Platform Communications – Unified Architecture</i> (Comunicações de Plataforma Aberta – Arquitetura Unificada)
PCP	<i>Planejamento e Controle da Produção</i>
PIMS	<i>Plant Information Management Systems</i> (Gerenciamento de Informações da Planta)
PMA	Planta de Manufatura Avançada
QMS	<i>Quality Management System</i> (Sistema de Gestão da Qualidade)
REST	<i>Representational State Transfer</i> (Transferência de Estado Representacional)
RFID	<i>Radio-Frequency Identification</i> (Identificação por Radiofrequência)
SAMOM	<i>Smart Adaptive Manufacturing Operations Management</i> (Gestão de Operações da Manufatura Inteligente e Adaptativa)
SCADA	<i>Supervisory Control And Data Acquisition</i> (Sistemas de Controle Supervisório e Aquisição de Dados)
SHM	Sistemas Holônicos de Manufatura
SMA	Sistemas Multiagentes
SOA	<i>Service-Oriented Architecture</i> (Arquitetura Orientada a Serviços)
SOAP	<i>Simple Object Access Protocol</i> (Protocolo de Acesso a Objetos Simples)
SQL	<i>Standard Query Language</i> (Linguagem de Consulta Estruturada)
SRM	Sistema Reconfigurável de Manufatura
UML	<i>Unified Modeling Language</i> (Linguagem de Modelagem Unificada)
WMS	<i>Warehouse Management System</i> (Sistema de Gerenciamento de Armazém)
XML	<i>Extensible Markup Language</i> (Linguagem de Marcação Extensível)

# GLOSSÁRIO

**Adaptabilidade:** principal e mais elevado estágio de maturidade tecnológica, no que se refere à Indústria 4.0, no qual os próprios sistemas tomam decisões, se otimizam e se reconfiguram de forma autônoma.

**Arquitetura convencional:** arquitetura de sistema tradicional de manufatura.

**Big data:** dados em grandes volumes.

**Dados historiados:** dados armazenados ao longo do tempo com seu respectivo instante de coleta.

**Descentralizado:** controlado e operado por subsistemas ou unidades sem a necessidade de estarem subordinados a apenas uma.

**Distribuído:** controlado e operado por várias unidades individuais, porém complementares, e livres para comunicação entre si.

**Escalável:** capaz de evoluir em volume e abrangência.

**Funcionalidade MES:** processadores de dados e informações setorizadas de G&C no MES.

**Gestão e controle:** tomadas de decisão, controle da produção e controle de sistemas de manufatura.

**Habilidade adaptativa:** capacidades do sistema em tomar decisões, com níveis de autonomia, eficiência, velocidade e abrangência em menor escopo que a adaptabilidade, adequados por suas operações.

**Habilidade preditiva:** capacidade do sistema em prever informações ou eventos, baseado em dados historiados.

**Heterárquico:** organização dos componentes do sistema na qual estes têm liberdade de controle e de comunicação entre si e não estão sujeitos a componentes subordinantes.

**Inteligência:** capacidade cognitiva, baseado na IA orientada por dados devidamente historiados e sintetizados.

**Manufatura:** conjunto de processos para transformar a matéria-prima em produto acabado.

**Manufatura 4.0:** manufatura apoiada por tecnologias e conceitos da I4.0.

**Manufatura tradicional:** manufatura que atua sob paradigmas das revoluções industriais anteriores à I4.0.

**Modular:** constituído por unidades autônomas autogerenciáveis.

**Organização:** empresas, instituições, entidades de administração pública ou organizações não governamentais.

**Orquestrar:** ato de garantir a execução do papel de cada elemento atuante nas operações da manufatura, de acordo com objetivos de negócio.

**Produto tangível:** produto físico, palpável, fabricado por processos de manufatura como usinagem, conformação, extrusão etc. Não inclui software, produtos digitais, projetos, serviços ou afins.

**Sistema de manufatura:** conjuntura organizada dos diversos e elementos atuantes nas operações da manufatura.

**String de busca:** descritor utilizado para busca de trabalhos científicos.



## Sumário

<b>INTRODUÇÃO</b> .....	<b>3</b>
1.1 OBJETIVO.....	7
1.2 OBJETIVOS ESPECÍFICOS .....	7
1.3 JUSTIFICATIVA .....	7
1.4 METODOLOGIA.....	9
1.5 ESTRUTURA DO TRABALHO .....	16
<b>FUNDAMENTAÇÃO TEÓRICA E ESTADO DA ARTE</b> .....	<b>17</b>
2.1 SISTEMAS E PROCESSOS DE MANUFATURA PARA A INDÚSTRIA 4.0 ..	17
2.2 GESTÃO E CONTROLE DE SISTEMAS DE MANUFATURA 4.0.....	22
2.3 SISTEMAS DE EXECUÇÃO DA MANUFATURA PARA A INDÚSTRIA 4.0 ..	28
2.4 TRABALHOS RELACIONADOS .....	32
<b>2.4.1 Direções de Pesquisa para Sistemas de Manufatura 4.0</b> .....	<b>32</b>
<b>2.4.2 Modelos e Arquiteturas de Sistemas de Manufatura 4.0</b> .....	<b>37</b>
<b>MODELO PARA GESTÃO E CONTROLE INTELIGENTES DA MANUFATURA</b> ...	<b>50</b>
3.1 ARQUITETURA PRINCIPAL .....	51
3.2 DISPOSITIVOS INTELIGENTES DE MANUFATURA.....	52
<b>3.2.1 Classes de Agentes</b> .....	<b>56</b>
3.3 SISTEMA DE EXECUÇÃO DA MANUFATURA .....	59
<b>3.3.1 Contexto de Gestão e Controle e Relações de Tempo</b> .....	<b>61</b>
3.4 COMUNICAÇÃO .....	63
<b>VALIDAÇÃO DO MODELO</b> .....	<b>67</b>
4.1 A PLANTA DE MANUFATURA AVANÇADA .....	67
4.2 CONTEXTO REPRESENTATIVO E CRITÉRIOS DE VALIDAÇÃO .....	73
4.3 INSTÂNCIA DO MODELO SAMOM NA PLANTA DE MANUFATURA AVANÇADA.....	74
<b>4.3.1 Sistema Multiagentes da Planta de Manufatura Avançada</b> .....	<b>75</b>
<b>4.3.2 Sistema de Execução da Manufatura da Planta de Manufatura     Avançada</b> 79	
<b>RESULTADOS E DISCUSSÕES</b> .....	<b>83</b>
<b>CONCLUSÃO</b> .....	<b>89</b>

6.1 OPORTUNIDADES PARA TRABALHOS FUTUROS.....	91
6.2 PRODUÇÃO ACADÊMICA GERADA.....	93
<b>REFERÊNCIAS.....</b>	<b>94</b>
<b>APÊNDICE A .....</b>	<b>97</b>
<b>APÊNDICE B .....</b>	<b>110</b>

*Neste capítulo são apresentados os objetivos gerais e específicos, justificativa, metodologia e estrutura deste trabalho.*

## **INTRODUÇÃO**

As fábricas atualmente operam seguindo procedimentos, tecnologias e padrões consolidados pelas revoluções industriais que ocorreram até então. As duas primeiras revoluções industriais estabeleceram processos de manufatura com atividades repetitivas manuais auxiliadas por máquinas (primeira revolução) e, principalmente, às características da divisão do trabalho e da produção em massa (segunda revolução) – permitidas pelas energias térmica e elétrica. A terceira introduziu a automação de processos, devido principalmente às tecnologias relacionadas a sistemas de controle computadorizados e eletrônicos. Contudo, tal automação se dá em nichos pouco ou não integrados à totalidade do sistema de manufatura e ainda atrelados aos padrões da produção em massa, em modelos hierarquizados de gerenciamento, herdados da segunda revolução industrial.

Tais sistemas, aqui denominados de sistemas de manufatura tradicionais, apresentam limitações na operação de processos como os de manufatura discreta<sup>1</sup> e

---

<sup>1</sup> Categoria de processos de manufatura que produzem unidades descontínuas de produtos, os quais podem ser de variadas configurações particulares. Exemplo: indústria de automóveis. Em contrapartida, o produto da manufatura contínua pode ser representado por um fluxo homogêneo e crescente. Exemplo: indústria petroquímica.

em batelada<sup>2</sup>. Estes são complexos e, por conta disso, limitam as possibilidades de diversidade e personalização de produtos (Wan *et al.*, 2021; Lu; Xu; Wang, 2020). Uma dessas limitações é relacionada à baixa flexibilidade, a qual diminui a capacidade do sistema produzir produtos variados e personalizados conforme solicitado pelo cliente (Kim *et al.*, 2020; Lu; Xu; Wang, 2020). Tal limitação se deve aos níveis de centralização e hierarquização destes sistemas, isto é, um elemento central controla elementos subordinados que compõem o sistema de manufatura, o que o torna rígido (Lu; Xu; Wang, 2020; Morgan *et al.*, 2021).

Os sistemas de manufatura necessitam de gestão e controle (G&C) para acompanhamento do desempenho, ajuste de parâmetros, tomadas de decisão, melhoria contínua dos processos ou monitoramento de características de qualidade do produto. Nos sistemas tradicionais de manufatura as atividades de G&C dependem elevadas quantidades de recursos em suas operações, principalmente humanos e de tempo. A diferença temporal entre a disponibilidade de informações acerca da produção, a interpretação destas e a consequente tomada de decisão demonstra tal fato. Outros exemplos que confirmam são a frequente realização de tarefas repetitivas por profissionais para resolução de problemas e a ausência do reaproveitamento, entre casos, do conhecimento adquirido em tais resoluções (Kim *et al.*, 2020).

Com o advento do mais abrangente e profundo processo de transformação da história da indústria, a quarta revolução industrial, comumente chamada de Indústria 4.0 (I4.0), as organizações passam a dispor de novas estratégias para seu desenvolvimento e sua continuidade competitiva no mercado. Os sistemas de manufatura integrados, inteligentes e distribuídos, isto é, controlados por vários componentes conectados – com determinada harmonia e capacidade cognitiva baseada em dados – e complementares, rompem paradigmas tradicionais da manufatura, trazendo novas abordagens para os sistemas de produção. Oportunidades, conceitos e tecnologias da I4.0, como conectividade ubíqua, disponibilidade de grandes volumes de dados (*big data*), Inteligência Artificial (IA),

---

<sup>2</sup> Categoria de processos de manufatura que geralmente possui etapas discretas e contínuas e resulta em lotes de produtos de mesma configuração. Exemplo: indústria de alimentos.

integração de sistemas e digitalização de processos, habilitam tais sistemas, aqui denominados de sistemas de manufatura 4.0 (Morgan *et al.*, 2021).

Estes sistemas têm, como último alvo, a adaptabilidade, isto é, a tomada de decisão autônoma e em níveis ideais de eficiência, velocidade e abrangência. (Schuh *et al.*, 2017) Contudo, até atingir este alvo, os sistemas de manufatura passarão por processos ao agregar, progressivamente, novas tecnologias e conceitos. Neste processo, os sistemas podem possuir níveis não ideais de certas habilidades como, por exemplo, preditivas e adaptativas. Tais habilidades são alcançadas principalmente devido a determinada inteligência do sistema, baseada na IA orientada por dados devida e adequadamente historiados e sintetizados (Lu; Xu; Wang, 2020).

Sistemas de manufatura 4.0 podem ser habilitados para formas mais eficientes de G&C como, por exemplo, planejamento e acompanhamento da produção mais assertivos, devido à compreensão mais holística do sistema e pelo fato destes estarem mais integrados e distribuídos. Além disso a capacidade do sistema, agora inteligente, de prever eventos e antecipar ações de melhoria, ou correção, para adaptarem seus processos eleva sua flexibilidade e confiabilidade. Dessa forma, expectativas como maior sustentabilidade, menores impactos negativos e custos podem ser atendidos.

A disponibilidade de *big data*, bem como a capacidade computacional e seus métodos para extração de conhecimento desses dados (Análise de Dados e IA), com apoio da ampla conectividade e comunicação (Integração de Sistemas e Interoperabilidade) permitidas pela emergência da Internet das Coisas (IoT, do inglês *Internet of Things*), são exemplos de oportunidades, conceitos ou tecnologias da I4.0 que têm o potencial de estabelecer G&C apropriados para sistemas de manufatura 4.0. Uma Arquitetura Orientada a Serviços (SOA, do inglês *Service-Oriented Architecture*) é útil para propiciar, como serviços disponíveis no sistema, atividades de G&C, de forma modular e distribuída. Esta se mostra interessante para viabilizar os sistemas de manufatura a alcançar habilidades adaptativas, diante da flexibilidade requerida na emergente I4.0.

A partir dessas informações formula-se a questão norteadora do problema: como estabelecer G&C para manufatura que assegure as habilidades de predição e

adaptabilidade requeridas pelos modelos de negócio emergentes, com sistemas de produção mais flexíveis para fornecer produtos mais personalizados?

Com isso, o presente trabalho visa investigar a seguinte hipótese: é possível, com apoio de tecnologias habilitadoras e conceitos da quarta revolução industrial, viabilizar uma arquitetura de G&C de processos integrados, distribuídos e adaptáveis para sistemas de manufatura discretos e em batelada.

Este trabalho propõe uma arquitetura com estas características, apresentada sob um modelo denominado Gestão de Operações da Manufatura Inteligente e Adaptativa – SAMOM (do inglês *Smart Adaptive Manufacturing Operations Management*).

No presente trabalho, as operações de manufatura tratam de processos envolvidos na transformação de matéria-prima em produto tangível. Sistema de manufatura se refere à conjuntura de todos os elementos atuantes em tais operações. Nestes processos estão incluídas as etapas, dentre as do ciclo de vida do produto e da cadeia de suprimentos, relacionadas ao chão de fábrica e seus métodos de fabricação (como usinagem, conformação, extrusão), procedimentos de transformação física ou química (como limpeza, aquecimento, montagem, fermentação) e procedimentos suplementares (como movimentações, transporte, armazenamento, embalagem). O problema da pesquisa em questão é delimitado a processos diretamente relacionados a tal produção, especificamente processos de manufatura discreta e em batelada, devido suas características particulares, que têm relações com a massiva demanda de produtos personalizados na I4.0. Portanto, estes processos definem o escopo que será tratado no presente trabalho. Processos de outras etapas, como projeto do produto, logística e distribuição entre as organizações da cadeia de suprimentos e entrega ao cliente não fazem parte deste escopo. Além disso, os processos de manufatura contínua não são objeto no presente trabalho, devido sua diferente natureza e formas de G&C.

## **1.1 OBJETIVO**

Propor uma arquitetura para gestão e controle integrados de sistemas de manufatura discretos e em batelada com características distribuídas e com habilidades preditivas e adaptativas.

## **1.2 OBJETIVOS ESPECÍFICOS**

I. definir o escopo da operação de um sistema integrado e distribuído de manufatura, caracterizando os conceitos de gestão, controle e adaptabilidade adotados;

II. apresentar um modelo de Arquitetura Orientada a Serviços, que permita a disponibilidade das habilidades propostas;

III. determinar características, padrões e critérios de relacionamento entre componentes e conceitos do modelo;

IV. determinar um conceito para análise inteligente e distribuída pelo sistema, baseada em dados.

V. validar a arquitetura proposta em uma prova de conceito, em planta piloto de manufatura 4.0.

## **1.3 JUSTIFICATIVA**

A G&C em sistemas de manufatura tradicionais é pouco eficiente em relação ao que se espera na I4.0. O procedimento de resoluções de problemas nos processos de manufatura – que envolve, por exemplo, a identificação de sinais fora de controle e causas raiz – além de ser reativo, depende muito tempo, é subordinado ao trabalho repetitivo humano e é específico para cada problema, sendo difícil a reutilização do conhecimento adquirido para solução de outros problemas. Esta baixa eficiência é devida ao fato de os sistemas tradicionais serem pouco habilitados para operação por meio de tecnologias digitais integradas. Logo, estes não possuem suficientes visibilidade e interpretação de dados do sistema, bem como previsibilidade e habilidades adaptativas. Por outro lado, para a I4.0 espera-se que, por meio do

conhecimento obtido a partir dos dados historicamente adquiridos, o monitoramento dos processos seja preditivo e automatizado e as soluções sejam reutilizáveis e proativas (He; Wang, 2018; Chang, 2017).

Um fato que reflete esta necessidade de G&C sob a perspectiva da I4.0 está no crescente número de pesquisas a isso relacionadas. A revisão sistemática de Bueno, Godinho Filho e Frank (2020), por exemplo, aponta um aumento de 4 para 42 artigos publicados que envolvem a relação entre a I4.0 e o Planejamento e Controle da Produção (PCP), entre os anos de 2015 e 2018. Mais recentemente, Shojaeinasab e colaboradores (2022) identificam um aumento de 16 para 228 artigos publicados, entre os anos de 2017 e 2021, sobre o emprego de tecnologias da I4.0 nos clássicos sistemas de gestão da manufatura, o Sistema de Execução da Manufatura (MES, do inglês *Manufacturing Execution System*).

O impacto financeiro e produtivo também é fator que promove a adequada G&C em sistemas de manufatura 4.0. Em uma pesquisa com 234 fornecedores de componentes automotivos, Arcidiacono e Schupp (2024) identificaram que é estatisticamente significativo que a adoção da manufatura inteligente é positivamente relacionada com melhorias no desempenho de custos. Tripathi e colaboradores (2022), na sua aplicação de um sistema de tomada de decisão orientada por dados, baseada em tecnologias que tornam a manufatura inteligente, em uma fábrica de máquinas de mineração, alcançaram melhorias de 66,67% na capacidade de produção. Por outro lado, Méndez e Rodriguez (2017), alcançaram apenas 10,7% com sua metodologia tradicional não baseada em tecnologias de I4.0, aplicada a uma linha de usinagem de uma peça automotiva.

A I4.0 é uma realidade e cada vez mais tomará lugar e substituirá a indústria atual, a qual se baseia na automação em conjuntos fechados e não integrados. Conseqüentemente, os sistemas de manufatura 4.0 serão progressivamente mais implementados. Estes naturalmente e conceitualmente adotam e integram tecnologias digitais para sua operação e têm como objetivos visibilidade e interpretação das informações de todo o sistema, para possibilitar a previsibilidade.

Diante, então, da necessidade de uma nova abordagem de G&C, a ser implementada sobre os sistemas de manufatura que evoluem de tradicionais para 4.0,



o presente trabalho promove as oportunidades, conceitos e tecnologias habilitadoras da I4.0. Estas apresentam potencial para ser o principal fator na atribuição de habilidades preditivas e adaptativas aos sistemas e no suprimento de tal necessidade.

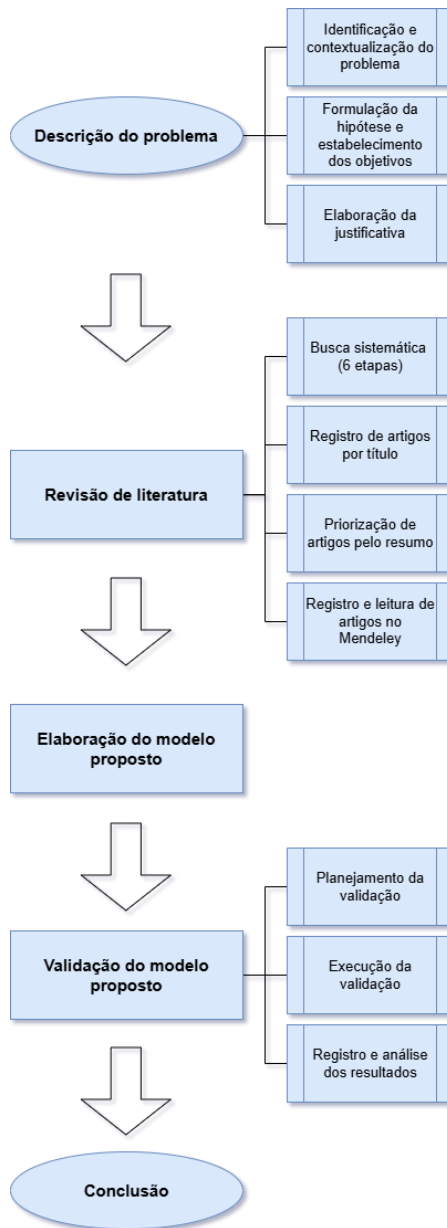
#### **1.4 METODOLOGIA**

De acordo com Marconi e Lakatos (2017) e Gil (2002), para que apresente resultados, verdadeiros, pertinentes e reproduzíveis, os trabalhos de pesquisa devem ser realizados sob um conjunto de atividades sistemáticas e racionais, que caracterizam o método científico. Segundo eles, para enquadrarem-se em tal método científico, de forma geral, as pesquisas se propõem a seguir as etapas de: identificação e descrição de um problema num conjunto de conhecimentos, formulação de hipóteses que indiquem a solução do problema, proposição de uma solução com auxílio de investigações conceituais e empíricas, comprovação da solução e, por fim, conclusão do trabalho com análise de resultados e indicações de correções e novos rumos. O presente trabalho submeteu-se à realização destas etapas.

Em concordância com tais critérios, a descrição da metodologia adotada no trabalho em questão é apresentada no fluxograma da Figura 1.1 e descrita nos tópicos a seguir.

- a) identificação e contextualização do problema: G&C de sistemas de manufatura na I4.0;
- b) formulação de hipótese: propõe uma solução para o problema baseada em tecnologias habilitadoras da indústria 4.0
- c) estabelecimento de objetivos
- d) determinação da justificativa: sustenta a investigação da hipótese para a solução do problema
- e) revisão da literatura: definição de escopos e caracterização de conceitos (sistemas de manufatura de I4.0, gestão, controle e adaptabilidade) e apresentação de trabalhos já realizados que atingem o problema em questão (abordando os conhecimentos utilizados e as lacunas que continuam).

Figura 1.1 Fluxograma da metodologia do trabalho



Fonte: Próprio Autor.

- f) apresentação de um modelo como uma solução para o problema: se desdobra em uma arquitetura e é baseado principalmente nas investigações apresentadas na revisão de literatura
- g) validação da arquitetura: realizada em uma planta piloto de manufatura habilitada para I4.0.

- h) apresentação e discussão de resultados: para análise da capacidade da arquitetura proposta. O trabalho é
- i) conclusão: avaliação do alcance de objetivos e indicação de perspectivas futuras.

O método descrito é contemplado em algumas classificações que o evidenciam também como científico, de acordo com Marconi e Lakatos (2017) e com Gil (2002). Quanto às razões que determinam sua realização, esta pesquisa é classificada como aplicada (ou tecnológica), pois visa diretamente obter contribuições práticas do conhecimento alcançado. Quanto aos seus objetivos, esta pesquisa é classificada como exploratória, pois visa proporcionar um maior entendimento de um problema a partir da investigação de uma hipótese, além de aprimorar conceitos para reuni-los numa proposta. No que se refere aos procedimentos técnicos utilizados (ou delineamento) na pesquisa, esta se enquadra em duas classes: bibliográfica e experimental. A primeira porque o presente trabalho é embasado e constituído, em parte, por um estudo a partir de fontes bibliográficas, principalmente artigos científicos. A segunda se deve ao fato de que aqui é proposta uma arquitetura, como objeto de estudo, sobre a qual variáveis e seus efeitos são analisados, inclusive por meio de uma implementação específica da proposta com o intuito de validá-la.

No que se refere à pesquisa sistemática, tanto a fundamentação teórica dos conceitos como, principalmente, o estado da arte, a partir de trabalhos relacionados, foram elaborados com base em referências obtidas por meio de uma pesquisa de literatura. Foram realizadas buscas de artigos, divididas em seis etapas, nas bases Science Direct, Engineering Village, IEEE Xplore e Google Acadêmico. Cada etapa corresponde a um determinado período na construção, amadurecimento e evolução do tema do projeto de dissertação.

Na primeira etapa foram utilizadas as seguintes palavras-chave: *service oriented architecture*, *soa*, *manufacturing* e *industry 4.0*. Na segunda etapa foi utilizada a seguinte *string* de busca: ("*service-oriented architecture*" OR *soa*) AND ("*industry 4.0*" OR "*fourth industrial revolution*" OR "*advanced manufacturing*" OR "*intelligent manufacturing*") AND (*management* OR *control* OR *prediction*). Na terceira etapa foram utilizadas as seguintes *strings*:

- a) (“*statistical process control*” OR “*statistical process monitoring*”) AND (“*service-oriented architecture*”);
- b) (“*statistical process control*” OR “*statistical process monitoring*”) AND (*review* OR *survey*).

Na quarta etapa foram utilizadas as seguintes *strings*:

- c) (*prediction*) AND (“*problem solving*” OR “*root cause*”) AND (“*integrated and distributed manufacturing systems*” OR “*advanced manufacturing*” OR “*fourth industrial revolution*”);
- d) (*prediction*) AND (“*problem solving*” OR “*root cause*”) AND (“*processes management*” OR “*process control*” OR “*process monitoring*”);
- e) (*prediction*) AND (“*process management*” OR “*process control*” OR “*process monitoring*”) AND (“*integrated and distributed manufacturing systems*” OR “*advanced manufacturing*” OR “*fourth industrial revolution* OR *industry 4.0*”).

Na quinta etapa foi utilizada a seguinte *string*: (*prediction*) AND (*adaptative* OR *optimization*) AND (*flexible*) AND (“*process management*” OR “*process control*” OR “*process monitoring*”) AND (“*advanced manufacturing*” OR “*smart manufacturing*” OR “*fourth industrial revolution* OR *industry 4.0*”). Na sexta e última etapa foram utilizadas as seguintes palavras-chave e *string*:

- a) *adaptability*, *adaptive* e *adaptive process control*;
- b) *flexibility*, *flexible* e *flexible system*;
- c) *intelligent manufacturing*, *smart manufacturing* e *advanced manufacturing*;
- d) *process management*, *process control* e *process monitoring*;
- e) *manufacturing system*;
- f) *operations management* e *manufacturing operations*;
- g) (*adaptability* OR *adaptive*) AND (*intelligence* OR *intelligent*) AND (*manufacturing* OR “*manufacturing system*”) AND (*survey* OR *review*).

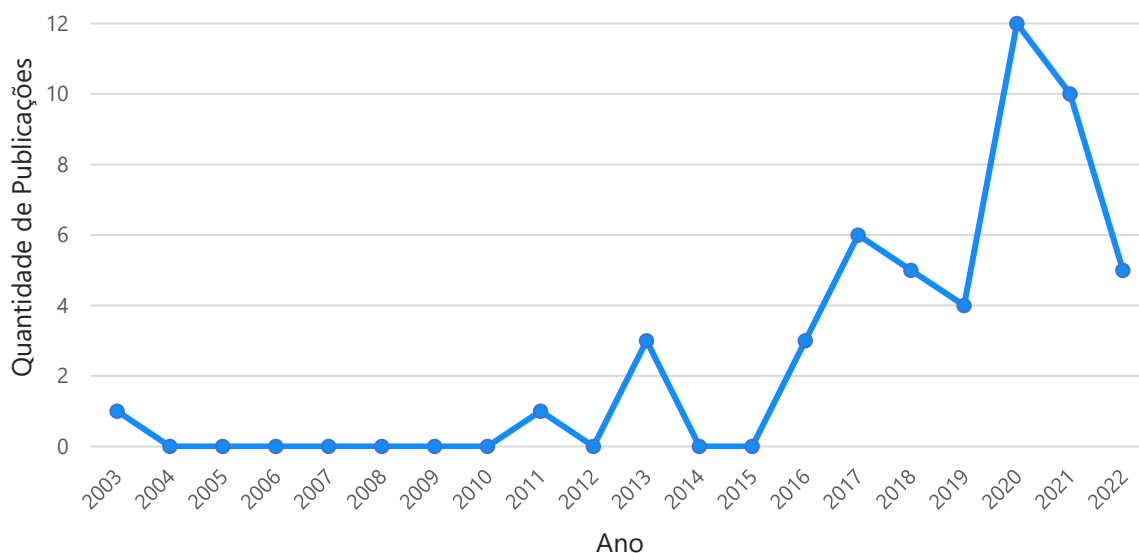
Após a realização de cada busca em cada etapa, trabalhos pertinentes iam sendo selecionados, com base em seu título, e registrados, em planilhas. Na primeira

etapa foram registrados 33 trabalhos, na segunda 59, na terceira 33, na quarta 35, na quinta 2 e, na sexta, 94. A partir da leitura do resumo destes trabalhos, alguns que se relacionam diretamente ao tema da presente pesquisa, iam sendo selecionados para serem lidos completamente. Estes foram sendo registrados no aplicativo Mendeley, para serem gerenciados por meio de coleções e etiquetas. À medida que os artigos foram sendo lidos, incluía-se no registro de planilhas, outros artigos referenciados por estes, além de artigos indicados por colegas de pesquisa e pelo orientador.

Foram registrados no fim, em planilha, um total de 256 artigos de periódico, de congressos ou capítulos de livro. O total de lidos completamente foi de 50 artigos. A Figura 1.2 apresenta a quantidade de publicações por ano, dentre estes 50. A série demonstra crescimento das publicações nos últimos anos, o que indica relevância do tema na atualidade. A queda nos de 2021 e 2022 pode ser interpretada como uma exceção devido a dois fatos:

- a) estes foram os anos em que a presente pesquisa foi realizada;
- b) trata-se de um período contemporâneo ao auge da pandemia da Covid-19.

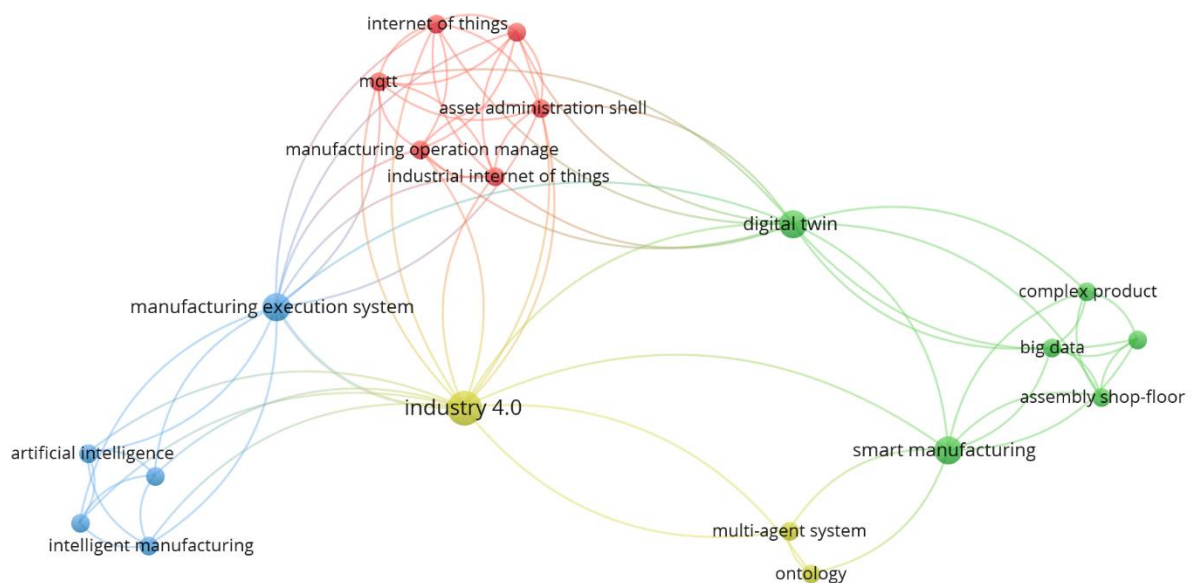
Figura 1.2 Série de publicações por ano



Fonte: Próprio Autor.

Dentre os 50 artigos lidos completamente, foram selecionados 14, todos de periódico, considerados como o estado da arte para a presente pesquisa. Estes estão identificados no Quadro 1.1 e são discutidos na seção 2.4. A Figura 1.3 apresenta uma rede semântica para coocorrência de palavras-chave nestes 14 trabalhos. A figura indica algumas palavras-chave centrais, como *Manufacturing Execution Systems* e *Smart Manufacturing*, que conectam às demais e confirmam o foco do presente trabalho. Este fato é pertinente ao que é apresentado na seção 2.

Figura 1.3 Rede de palavras-chave



Fonte: Próprio Autor.

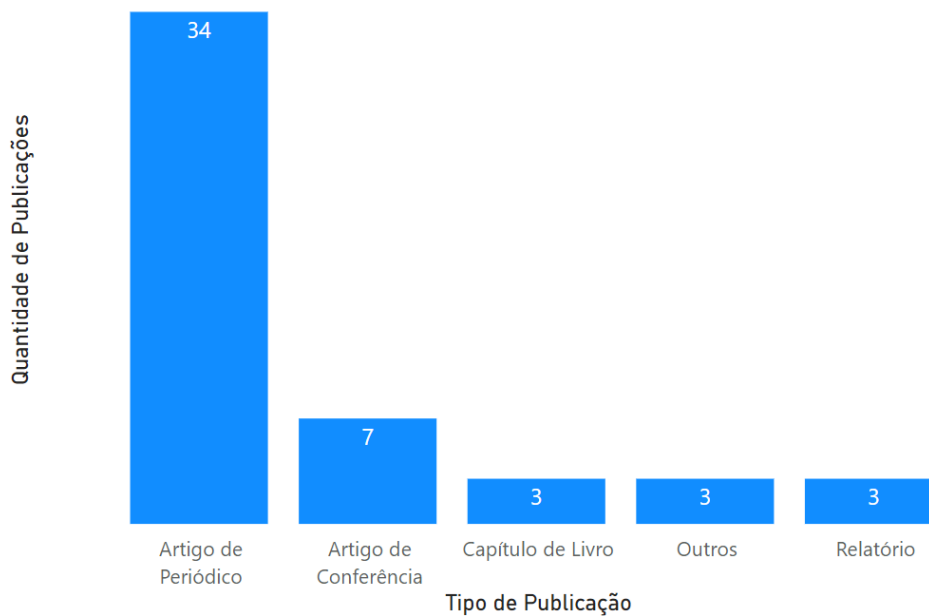
Em geral priorizou-se artigos de periódicos a artigos de conferências e capítulos de livro, como pode ser percebido na Figura 1.4, que apresenta as classes dos 50 trabalhos lidos. Dentre estes, além dos 14 selecionados como estado da arte, alguns outros trabalhos foram citados como referências teóricas no presente trabalho. Fora destes 50, como referência teórica foram citados livros, normas e publicações de instituições industriais.

Quadro 1.1 Trabalhos considerados como estado da arte

Referência	Síntese
Kruger e colaboradores (2011)	Propõem uma arquitetura orientada a agentes e com algoritmos inteligentes para regulação de processos de usinagem.
Tang e colaboradores (2017)	Fazem uma revisão de arquiteturas baseadas em agentes para controle da manufatura e propõem e experimentam a sua, assistida por nuvem.
Zhuang, Liu e Xiong (2018)	Propõem uma estrutura para G&C inteligente da produção, baseada em GD, para linhas de montagem de produtos complexos e a testa em uma fábrica de montagem de satélites.
Ding e colaboradores (2020)	Fazem uma pesquisa para identificação do estado atual da manufatura orientada por IA. Resume as principais tecnologias de IA e discute cenários de aplicação com relação diagnóstico de falhas, previsão de vida útil restante e inspeção de qualidade.
Bueno, Godinho Filho e Frank (2020)	Realizam uma revisão sistemática para investigar a influência sobre o PCP de cinco elementos da I4.0 (IoT, CPS, Análise de <i>Big Data</i> , IA e Manufatura Aditiva).
Kim e colaboradores (2020)	Propõem um sistema de manufatura inteligente usando máquinas com agentes inteligentes, em um SMA, e aprendizado por reforço para autonomia de tomada de decisão.
Wan e colaboradores (2020)	Pesquisam sobre a implementação da Ia na manufatura personalizada e propõem uma arquitetura baseada em dispositivos inteligentes para sistemas de manufatura flexíveis.
Usuga Cadavid e colaboradores (2020)	Fazem um levantamento de estado da arte na literatura no que se refere ao PCP auxiliado por ML.
Jaskó e colaboradores (2020)	Fazem uma revisão de padrões, metodologias e ferramentas no desenvolvimento de MES influenciado pela I4.0, identificando tendências atuais e futuras.
Lu, Xu e Wang (2020)	Realizam uma revisão do cenário atual de padrões de automação de manufatura, com foco em processos e sistemas de manufatura integrados para personalização em massa.
Arm e colaboradores (2021)	Discutem a estrutura, componentes, submodelos e protocolos de comunicação de um método de gestão inteligente da produção, que utiliza ativos inteligentes.
Morgan e colaboradores (2021)	Fazem uma pesquisa de exploração do estado da arte em inteligência e controle de máquina distribuído e descentralizado, além de propor um conceito de sistemas reconfiguráveis de manufatura.
Buckhorst, Grahn e Schmitt (2022)	Analisam a literatura para identificar componentes de arquiteturas flexíveis para sistemas de manufatura personalizada e propõem sua arquitetura, baseada num paradigma para sistemas de montagem.
Shojaeinasab e colaboradores (2022)	Realizam uma revisão sistemática acerca de MES inteligentes e propõem um modelo conceitual para realização de tarefas de MES com capacidade de previsão e adaptabilidade.

Fonte: Próprio Autor.

Figura 1.4 Quantidade de publicações por tipo



Fonte: Próprio Autor.

## 1.5 ESTRUTURA DO TRABALHO

Este projeto de dissertação está organizado como é descrito a seguir. Neste Capítulo foi apresentada uma introdução para contextualização, descrição e delimitação do problema, bem como a apresentação de uma hipótese para solução deste, além dos objetivos, justificativa e metodologia do trabalho. No Capítulo 2 é apresentada uma revisão de literatura composta por referencial teórico relacionado a algumas definições, estado da arte acerca de alguns conceitos e tecnologias, além de trabalhos relacionados. No Capítulo 3 é proposto e descrito um modelo, que se desdobra em uma arquitetura, para solução do problema. No Capítulo 4 é apresentada a descrição da aplicação do modelo proposto, para validação do mesmo, no que se refere aos materiais, métodos e infraestrutura utilizados. O Capítulo 5 apresenta os resultados alcançados na referida validação, bem como as discussões destes. O Capítulo 6, e também último, apresenta a conclusão, as considerações futuras e a produção acadêmica deste trabalho.



*Nesta seção são discutidos e definidos alguns conceitos importantes para o entendimento do presente trabalho, baseados em um referencial teórico, como também é trazido o estado da arte relacionado ao presente tema.*

## **FUNDAMENTAÇÃO TEÓRICA E ESTADO DA ARTE**

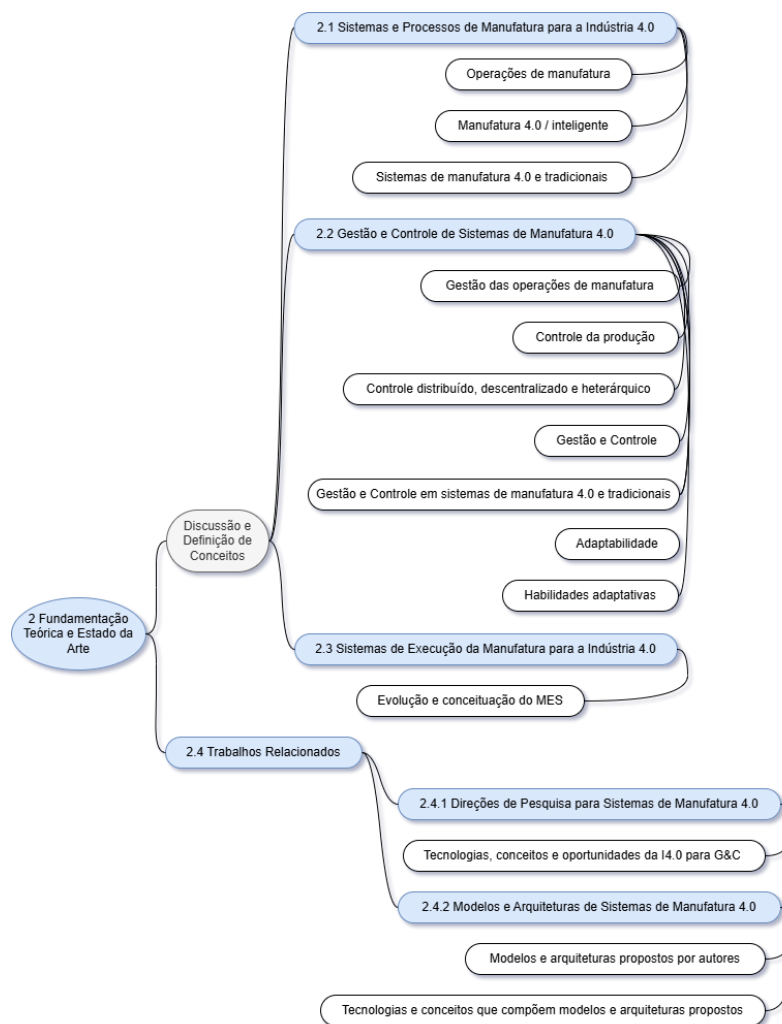
O mapa mental da Figura 2.1 resume a estrutura deste capítulo. Na subseção 2.1 são discutidos os conceitos de operações de manufatura, manufatura 4.0 e sistemas de manufatura 4.0. Na subseção 2.2 são discutidos os conceitos de gestão e controle de sistemas de manufatura e de habilidades adaptativas. Na subseção 2.3 são discutidos os conceitos de sistemas de execução da manufatura e de sistemas de gestão das operações de manufatura. Por fim, são apresentados na subseção 2.4 trabalhos relacionados que contribuem com o estado da arte em relação a arquiteturas para sistemas de manufatura 4.0 e a tecnologias habilitadoras que as compõe, além de indicar lacunas e direções de pesquisa.

### **2.1 SISTEMAS E PROCESSOS DE MANUFATURA PARA A INDÚSTRIA 4.0**

Stevenson (2021) define operações como processos, de uma empresa, que fornecem serviços ou criam itens tangíveis (bens) ao envolver a transformação ou conversão de insumos em produtos. Entretanto, além de empresas, qualquer outro tipo de organização também utiliza de operações para realizar serviços ou oferecer produtos. Além disso, também existem operações que resultam em produtos

intangíveis. No escopo tratado por Stevenson (2021), as operações e as cadeias de suprimentos são intrinsecamente ligadas e esta última se trata da sequência de organizações envolvidas na produção e entrega de um serviço ou bem, por meio de suas instalações (armazéns, fábricas, escritórios, centros de distribuição, lojas etc.), funções e atividades (previsão, compras, gerenciamento de estoque, gerenciamento de informações, garantia de qualidade, programação, produção, distribuição, entrega e atendimento ao cliente).

Figura 2.1 Mapa da fundamentação teórica e estado da arte



Fonte: Próprio Autor.

Lu, Xu e Wang (2020) definem a manufatura como a etapa para converter a matéria-prima no produto acabado. De acordo com os autores, na I4.0 os processos

de manufatura concentram-se na produção discreta, de um produto especificado, que pode envolver diversos métodos de fabricação, com processos que precisam alcançar integração suficiente (por meio da interoperabilidade) e, conseqüentemente, a requerida flexibilidade dos sistemas de I4.0. Em concordância com os autores, o presente trabalho se baseia no fato de que a personalização de produtos (apoiada da alta flexibilidade dos sistemas, possibilitada pelos conceitos e tecnologias da I4.0) pode promover maior disseminação de processos de manufatura discreta ou semelhantes (por exemplo, manufatura aditiva e processos de batelada). Entretanto, cabe destacar que a manufatura contínua também é contemplada pelos avanços da I4.0.

De acordo com Kusiak (2018), não há uma definição para manufatura inteligente genericamente aceita. Ele afirma que a manufatura inteligente integra ativos de manufatura a sensores, plataformas de computação, tecnologia de comunicação, modelagem de dados, controle, simulação e engenharia preditiva. Para o autor, tal conjunto é implementado a partir de tecnologias habilitadoras da indústria 4.0, como Sistemas Ciberfísicos (CPS, do inglês *Cyber Physical Systems*), IoT, computação em nuvem, SOA, IA e Ciências de dados. Phuyal, D. Bista e R. Bista (2020) afirmam que a manufatura inteligente integra tecnologias relacionadas à fabricação, computação, virtualização, conectividade e análise de dados. De acordo com eles, a I4.0 amplia a implementação destas tecnologias para o alcance de melhor custo-benefício, economia de tempo, configuração fácil e rápida, resposta rápida à demanda de mercado e flexibilidade.

Lu, Xu e Wang (2020) afirmam que o termo manufatura inteligente é controverso devido ao fato de estar sendo visto como um conceito que descreve manufatura com inteligência avançada, além de se tratar de um paradigma específico de manufatura. Os autores conceituam manufatura inteligente como operações integradas, colaborativas e responsivas capazes de responder em tempo real às mudanças nas demandas na fábrica, cadeia de suprimentos e necessidades do cliente, a partir do gerenciamento baseado em dados e da execução, de tais operações, apoiadas pela utilização de tecnologias avançadas de detecção, modelagem, simulação e análise. Os autores ainda afirmam que a manufatura avançada tem papel fundamental na

mudança da pirâmide de automação, convencional da terceira revolução industrial, para uma rede de manufatura auto-organizada e com capacidades cognitivas.

Ding e colaboradores (2020) definem a manufatura inteligente, como auto-evolutiva e dotada de capacidade cognitiva, independente de tecnologias atuais ou que possam contribuir futuramente. Os autores apontam a IA como a principal tecnologia que suporta a manufatura inteligente.

Wang e colaboradores (2021) realizaram uma revisão sistemática com o intuito de elucidar as diferenças entre os termos *Smart Manufacturing* e *Intelligent Manufacturing*. Ambos os termos possuem apenas uma tradução para determinadas línguas. No português a tradução única é manufatura inteligente. De acordo com os autores, ambos descrevem a evolução da manufatura facilitada por tecnologias avançadas de informação e comunicação, contudo, trata-se de paradigmas paralelos que evoluíram de forma independente e foram objetos de diferentes grupos. Os autores concluíram que enquanto *Smart Manufacturing* é encontrado mais frequentemente junto a conceitos de I4.0, orientação a dados e *big data*, *Intelligent Manufacturing* é encontrado com mais frequência junto a conceitos de algoritmo, otimização, orientação a agentes e arquitetura de IA. Eles ainda afirmam que o desenvolvimento da digitalização, rede e inteligência na manufatura é comum para ambos.

Diante da ausência de uma definição objetiva e de termos consolidados na literatura, o presente trabalho opta por utilizar o termo “manufatura 4.0”, sintetizando os conceitos que melhor lhe fazem sentido. “manufatura 4.0” é aqui definida como uma manufatura apoiada por tecnologias habilitadoras da I4.0, como análise de dados, IA, integração de sistemas, computação na nuvem e na borda, conectividade ubíqua (principalmente IoT) e SOA, para o alcance de habilidades preditivas e adaptativas (para resposta rápida às demandas de mercado), flexibilidade e ágil disponibilidade de produtos personalizados. Os termos “inteligência” e “inteligente”, no presente trabalho, são utilizados para se referir apenas ao atributo específico de capacidade cognitiva, baseado na IA orientada por dados devidamente historiados e sintetizados.

A literatura também apresenta diferentes abordagens para “sistemas” de manufatura da I4.0. Phuyal, D. Bista e R. Bista (2020), por exemplo, definem seu conceito de Sistema de Manufatura Inteligente como a digitalização de todas as partes de um sistema de manufatura. Para os autores, este dispõe de interoperabilidade, controle e monitoramento em tempo real, flexibilidade e adaptabilidade. Morgan e colaboradores (2021) trazem o conceito de Sistema Reconfigurável de Manufatura (SRM) como um sistema projetado para ajuste rápido do funcionamento e da capacidade da produção, por meio da readequação ou alteração dos ativos de manufatura, para adaptação à novas situações. Segundo os autores, para dimensionar a capacidade e a capacidade de produção, o SRM permite a modificação rápida de suas funções e processos por meio de *hardware* e *software* adaptáveis.

De acordo com Lu, Xu e Wang (2020), tradicionalmente os sistemas de manufatura automatizados renunciam a flexibilidade para alcançar maior produtividade. Para os autores, estes sistemas rígidos e totalmente automatizados não são capazes de corresponder à personalização em massa, a qual é um alvo da I4.0. Portanto, os autores afirmam que uma rede de manufatura auto-organizada será um padrão para sistemas de manufatura, o que possibilita uma automação mais flexível e produtiva para atender a personalização em massa.

Este trabalho utiliza o termo “sistema de manufatura 4.0” e o define como a conjuntura organizada dos diversos e elementos atuantes nas operações da manufatura apoiada por tecnologias e conceitos da I4.0 (manufatura 4.0). É de fundamental interesse para o presente trabalho diferenciar tais sistemas de manufatura 4.0 de sistemas tradicionais. Denomina-se então como “sistemas tradicionais de manufatura” aqueles os quais o conjunto dos diversos elementos das operações de manufatura atuam sob paradigmas das revoluções industriais anteriores à I4.0. No caso da terceira revolução industrial, as operações de manufatura são apoiadas pela automação, mas geralmente não favorecem uma integração adequada do sistema. Nos paradigmas de primeira e segunda revoluções industriais, as operações de manufatura são baseadas em atividades manuais repetitivas.

## 2.2 GESTÃO E CONTROLE DE SISTEMAS DE MANUFATURA 4.0

Para Stevenson (2021), a gestão de processos possui papel central na gestão das operações e cada organização determina seu próprio escopo desta. Ele afirma que a gestão das operações pode envolver o projeto de produtos e serviços, o projeto de sistemas de trabalho, o planejamento de localização e instalações, o gerenciamento de tecnologia e a gestão da qualidade dos produtos ou serviços.

Em sua revisão de literatura para análise histórica da gestão das operações, Rungtusanatham e colaboradores (2003) determinaram cinco principais áreas para gestão das operações:

- a) *Just-in-Time*, que trata dos processos de manufatura e visa a melhoria sua melhoria contínua, bem como de produtos;
- b) Gerenciamento de Tecnologia, que trata das tecnologias implementadas nos sistemas de manufatura;
- c) Estratégia de Operações, que trata da tomada de decisão e estratégias de negócio;
- d) Gestão da Qualidade, que visa aumentar produtividade e a satisfação dos clientes, por meio do controle e monitoramento da qualidade nos processos de manufatura;
- e) Gestão da Cadeia de Suprimentos, que lida com o fluxo de produtos e informações e com os relacionamentos entre os componentes da cadeia.

No que se refere à manufatura na I4.0, tais áreas ainda são importantes e possuem fundamental papel na gestão das operações abordada no presente trabalho.

Segundo Fettermann e colaboradores (2018), em torno de 1980, a pesquisa relacionada a gestão das operações passou de uma abordagem baseada em modelo, com a utilização de métodos de otimização ou simulação, para pesquisas empíricas, orientadas por opinião ou percepção, com o intuito de melhor compreensão de questões como Estratégia de Operações, *Just-in-Time* ou Gestão da Qualidade. Para a era da I4.0, eles afirmam que diferentes combinações de tecnologias podem viabilizar novas estratégias que influenciam na gestão das operações, como uma manufatura mais ágil, maior flexibilidade de processos e personalização em massa. As abordagens dos autores acima, sobre Gestão das Operações, são não somente

pertinentes como se complementam. O presente trabalho define Gestão das Operações, delimitada à manufatura, como a gestão de processos de sistemas de manufatura, que envolve atividades das áreas de melhoria contínua de processos, gerenciamento de tecnologias do sistema, tomada de decisão, gestão da qualidade de produtos e gestão do fluxo de dados, informações e produtos.

Para Buckhorst, Grahn e Schmitt (2022), o PCP abrange a definição (planejamento) e a execução (controle) de cronogramas de produção. Para Bonney (2000), PCP tem por função a determinação do que se deve produzir, bem como da quantidade e do tempo para produção, além do monitoramento desta para verificação do cumprimento de tal determinação. Apesar de ser um paradigma tradicional, o PCP naturalmente possui características comuns aos interesses da manufatura 4.0. Segundo Bueno, Godinho Filho e Frank (2020), por exemplo, o PCP visa adaptar o sistema de manufatura à novas situações e oportunidades operacionais e solicitações complexas de clientes. Além do mais, para estes autores, devido a dinâmica de mudanças na indústria, a gestão das operações e, conseqüentemente a função PCP, necessitam incluir uma perspectiva integradora e que evolua para um novo contexto (neste caso a I4.0).

Usuga Cadavid e colaboradores (2020) definem o PCP, no contexto da I4.0, como a função que determina a quantidade para produção, com o intuito de alcançar objetivos comerciais como lucratividade e produtividade, e o controle dos processos de produção, o qual permite o monitoramento em tempo real da produção e a personalização do produto. Para os autores, tal controle e monitoramento de processos consiste no ajuste inteligente de parâmetros de funcionamento de recursos físicos de manufatura, baseado na análise constante dos dados de chão de fábrica, com o objetivo de otimizar ou controlar automaticamente tais recursos físicos.

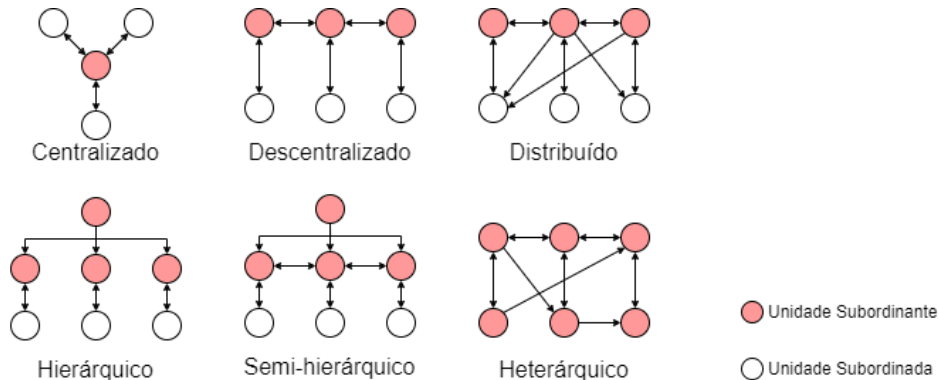
Morgan e colaboradores (2021) trazem uma discussão que permite uma comparação entre o controle tradicional, o qual os autores se baseiam na teoria de controle e no padrão ISA-88<sup>3</sup>, e o controle em sistemas de manufatura da I4.0. Os autores diferenciam os conceitos de controle distribuído, centralizado e

---

<sup>3</sup> Norma da Sociedade Internacional de Automação (ISA, do inglês *International Society of Automation*), publicada em 1995, que trata do controle de em processos de quantidades finitas de insumos e por um período finito de tempo – Controle de Lote (ou Batelada) (ISA, 1995).

descentralizado (Ver Figura 2.2). Para eles, no controle centralizado, uma unidade central controla a tecnologia localmente e atua sob estruturas hierárquicas, na qual alguns componentes respondem e se comunicam apenas à autoridade de determinado componente. No controle descentralizado, segundo os autores, várias unidades de igual importância são capazes de controlar ao operar paralelamente sob estruturas hierárquicas, heterárquicas - na qual os componentes não estão restritos a autoridade de determinado componente e têm liberdade para se comunicar com os demais – e semi-hierárquicas – na qual os componentes têm liberdade para se comunicar com componentes que operam paralelamente, mas devem responder a autoridade de determinado componente. O controle descentralizado também possui, de acordo com os autores, a característica de escalabilidade, que permite a adição ou remoção de componentes ou recursos no sistema. No controle distribuído, para os autores, várias unidades conectadas formam o sistema e são capazes de controlar tecnologia local e globalmente, sob estruturas heterárquicas e semi-hierárquicas.

Figura 2.2 Comparativo de controle centralizado, descentralizado e distribuído e estruturas hierárquicas, semi-hierárquicas e heterárquicas



Fonte: Adaptado de Morgan e colaboradores (2021).

Buckhorst, Grahn e Schmitt (2022) discutem sobre abordagens hierárquicas e heterárquicas sob o ponto de vista de relações de ordenação dos elementos em sistemas de manufatura. Os autores afirmam que numa hierarquia, sistemas menores (subsistemas) são subordinados ou pertencem a outros sistemas e formam uma estrutura lógica, na qual os componentes estão em ordem de classificação. Numa heterarquia, segundo os autores, cada componente é livre para cooperar com outros componentes ao redor, sem necessidade de orientação ou permissão de um



componente subordinante, e podem formar subsistemas autônomos organizados horizontalmente.

Arm e colaboradores (2021) afirmam que a I4.0 baseia-se no controle distribuído e descentralizado, de forma que os processos são executados sem uma unidade de controle central e a tomada de decisão é distribuída entre as unidades do sistema de manufatura. Além disto, os autores estabelecem dois níveis de execução da manufatura: o primeiro a partir do controle da produção, relacionada às operações de manufatura, e o segundo a partir do controle convencional de dispositivo.

O presente trabalho utiliza das abordagens de controle trazidas acima para convergir a definição deste conceito em dois níveis, semelhante à abordagem de Arm e colaboradores (2021). O primeiro nível, baseado na função PCP, trata do controle da produção, que visa o monitoramento das operações de manufatura, bem como a atuação do sistema para garantia da execução de tais operações conforme o determinado. O segundo nível abrange o convencional controle de máquinas, como apresentado por Morgan e colaboradores (2021). Contudo, o foco neste segundo nível é a determinação de sistemas distribuídos, descentralizados e heterárquicos de manufatura, a partir do controle dos componentes do sistema. Sendo assim, estabelece-se também para a presente pesquisa os conceitos:

- a) sistema distribuído: controlado e operado por várias unidades individuais, porém complementares, e livres para comunicação entre si.
- b) sistema descentralizado: controlado e operado por subsistemas ou unidades sem a necessidade de estarem subordinados a apenas uma.
- c) estrutura heterárquica: organização dos componentes do sistema na qual estes têm liberdade de controle e de comunicação entre si e não estão sujeitos a componentes subordinantes.

As características de modularidade – unidades autônomas autogerenciáveis – e de escalabilidade - capacidade de evoluir em volume e abrangência – também são conceitos essenciais para o controle de sistemas de manufatura 4.0 aqui preconizados.

Com o estabelecimento dos conceitos de gestão das operações de manufatura e de controle em sistemas de manufatura, convém, no escopo da presente pesquisa, unificá-los em apenas um termo, Gestão e Controle (G&C), devido a

complementaridade de suas atividades no que diz respeito ao papel de cada conceito na manufatura 4.0.

A G&C apresenta abordagens e características de implementação diferentes nos sistemas de manufatura 4.0 em relação aos tradicionais. Phuyal, D. Bista e R. Bista (2020) afirmam que os sistemas tradicionais serão substituídos pelos sistemas com tecnologias da I4.0, os quais apresentam maior capacidade de lidar com a flexibilidade para customização do produto e com as variações repentinas nas operações de manufatura. Zhuang, Liu e Xiong (2018) afirmam que a G&C em sistemas de manufatura tornou-se uma das questões mais abordadas por acadêmicos e empresas. Para eles, a G&C está em evolução, da automação e digitalização para a métodos inteligentes, com o advento da I4.0

Segundo Kim e colaboradores (2020), no que se refere à tomada de decisão, por exemplo, nos sistemas tradicionais o procedimento é centralizado e repetitivo, no qual decisões atuais são semelhantes às decisões anteriores, além de ser focado na solução mais próxima possível do que é considerado ótimo, e não na agilidade e adaptação em tempo real. Por outro lado, nos sistemas de manufatura da I4.0, a tomada de decisão deve oferecer capacidade de resposta flexível e em tempo real com informações locais, além de aprendizado e adaptação diante de um cenário dinâmico. O Quadro 2.1 apresenta algumas características, trazidas pelos autores, que diferenciam os sistemas de produção personalizados dos tradicionais.

Quadro 2.1 Características dos sistemas de produção personalizados e dos tradicionais

<b>Características</b>	<b>Sistemas tradicionais de manufatura</b>	<b>Sistemas de produção personalizados</b>
<b>Características do produto</b>	Padronizado	Personalizado
<b>Problema de tomada de decisão</b>	Repetitivo e semelhante	Único e novo
<b>Ciclo de tomada de decisão</b>	Periódico	Tempo real
<b>Estratégia de manufatura</b>	Valor	Flexibilidade
<b>Finalidade da manufatura</b>	Produtividade	Sustentabilidade
<b>Atributo do sistema</b>	Centralização	Descentralização

Fonte: Adaptado de Kim e colaboradores (2020).

A abordagem de Wilhelm e colaboradores (2020) contribui também na

compreensão da diferença entre tais sistemas. Os autores realizam uma revisão sobre a ciência de dados no monitoramento estatístico de processos para o controle da qualidade de produtos. Eles afirmam que soluções são apresentadas em algoritmos individuais por grupos de profissionais. Estas são impedidas de serem reutilizadas, o que leva a esforços igualmente elevados para criação de próximas soluções necessárias, ou a desgastante tarefa de criar e testar diferentes soluções por tentativa e erro. Para a I4.0, espera-se que o conhecimento, obtidos a partir de dados, seja reutilizado em novas e melhores soluções que impactam na qualidade do produto.

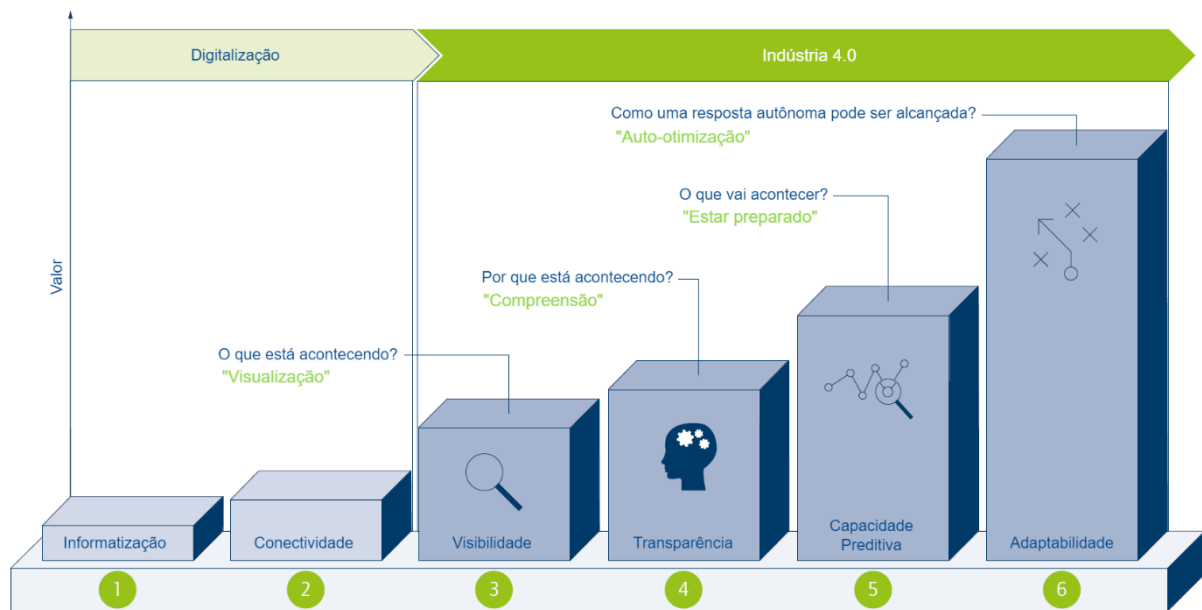
Morgan e colaboradores (2021) definem a adaptabilidade como a capacidade do sistema de manufatura em responder a mudanças relacionadas a quantidade e características de produtos. Para Jaskó e colaboradores (2020), o objetivo da adaptabilidade é atribuir ao sistema de manufatura a capacidade da melhor tomada de decisão possível no menor tempo possível, com base em dados. Eles afirmam que tal capacidade pode variar de simples a altamente complexa.

Para tornar uma empresa ágil no contexto da I4.0, capaz de se adaptar às dinâmicas condições de mudança e atender às necessidades flexíveis do cliente, a Acatech (SCHUH *et al.*, 2017) desenvolveu um estudo que propõe um modelo de transformação baseado em uma série de estágios de maturidade da I4.0. Este modelo visa fornecer um meio para identificar o atual estágio de maturidade na I4.0 de empresas e definir procedimentos para estas alcançarem estágios mais altos. Na Figura 2.3 são apresentados estes estágios, sendo os dois primeiros requisitos básicos para os níveis de I4.0, agrupados como digitalização, e os restantes os reais estágios de I4.0, dentre os quais as empresas podem escolher estrategicamente seu nível alvo de maturidade. Segundo os autores, o último, principal e mais elevado estágio do modelo, o de adaptabilidade, visa a adaptação contínua. Nesta, os sistemas têm o papel de tomar decisões automatizadas para adaptar o sistema, em tempo real, a um ambiente de negócios em constante mudança.

De acordo ainda com Schuh e colaboradores (2017), em determinados casos pode ser melhor apenas automatizar processos individuais, isto é, o grau de adaptabilidade do sistema para determinada implementação depende de sua complexidade bem como da relação custo-benefício. Tal afirmação permite analisar a viabilidade de implementar algumas habilidades adaptativas individuais em processos

de manufatura ao invés da adaptabilidade total, a qual pode ser vista como uma realidade distante, pelo fato de buscar autonomia pelo sistema de manufatura em decisões amplas, além de ser o principal alvo da I4.0 (pelo modelo apresentado pelos autores).

Figura 2.3 Estágios de maturidade tecnológica da I4.0



Fonte: Adaptado de Schuh e colaboradores (2017).

Para o presente trabalho, a partir das definições de adaptabilidade trazidas acima, principalmente em concordância com Schuh e colaboradores (2017), um sistema de manufatura 4.0 pode possuir “habilidades adaptativas”. Este termo é definido como as capacidades do sistema em tomar decisões, com níveis de autonomia, eficiência, velocidade e abrangência determinados por suas operações. Enquanto que a adaptabilidade visa a totalidade do sistema e condições ótimas, as habilidades adaptativas aqui podem ser estabelecidas em menor escopo, adequado às operações.

### 2.3 SISTEMAS DE EXECUÇÃO DA MANUFATURA PARA A INDÚSTRIA 4.0

Chen e Voigt (2020) afirmam que, devido a uma referência contemporânea (por volta da década de 1970 à de 1990) de obter um gerenciamento total da empresa,

houve uma evolução em sistemas de gerenciamento, partindo do sistema de Planejamento de Necessidade de Materiais (MRP, do inglês *Material Requirement Planning*) – que integra previsão, programação mestre e compras –, passando pelo MRP II (uma nova abordagem do MPR, que visa um controle mais detalhado de processo) e também por sistemas de manufatura integrada por computador (que traziam abordagens sistemáticas para apoio de empresas de manufatura), até o Planejamento de Recursos Empresariais (ERP, do inglês *Enterprise Resource Planning*) – que são voltados para a integração de funções organizacionais. De acordo com os autores, mesmo com tal evolução, a necessidade de gestão das operações de manufatura no chão de fábrica ainda não foi atendida com o nível de granularidade e velocidade requeridos, o que motivou o surgimento, na década de 1990, do MES.

De acordo com Shojaeinasab e colaboradores (2022), o MES faz parte da tradicional pirâmide da automação, com o papel de conectar a camada de gerenciamento de alto nível, a do ERP, com a camada de controle de dispositivos de chão de fábrica. Segundo os autores, o objetivo do MES é auxiliar os profissionais na tomada de decisão para gestão das operações de manufatura. Ainda de acordo com os autores, o MES pode ser visto como um dos principais facilitadores da manufatura 4.0 por, especialmente, servir como base para implementação dos conceitos da I4.0 nas organizações. Para os autores, os novos MES para a I4.0 devem ser inteligentes, adaptáveis e desenvolvido sobre SOA, para programação ideal e reprogramação da produção, tomada de decisão em tempo real, detecção e diagnóstico de falhas de qualidade, alocação de recursos, coordenação das cadeias de suprimentos, coleta e análise de dados, controle de processos e manutenção preditiva.

Morgan e colaboradores (2021), ao descrever o padrão ISA-95<sup>4</sup>, que trata da integração entre sistemas corporativos e de controle, afirmam que o MES e o ERP têm por responsabilidade a programação da produção, a gestão das operações de manufatura e o planejamento de recursos, enquanto os Sistemas de Controle Supervisório e Aquisição de Dados (SCADA, do inglês *Supervisory Control And Data Acquisition*) atuam com o controle em nível de máquina. No entanto, para o presente trabalho, o foco é apenas o MES, por sua natural capacidade de gerenciamento da

---

<sup>4</sup> Norma da ISA que propõe uma interface para padronização da integração entre sistemas de controle e corporativos – Integração do Sistema de Controle Corporativo (ISA, [2023?]).

produção. O ERP visa uma gestão corporativa do negócio, alcançando por exemplo, faturamento, orçamento, contabilidade ou marketing em uma empresa, e para isso tem como uma de suas fontes de dados e informações o MES.

Ainda de acordo com Morgan e colaboradores (2021), o MES possui gerenciamento centralizado, a partir de transações orientadas a serviços, por meio de comunicação em rede direta a máquinas ou Interfaces Homem-Máquina (IHM) no chão de fábrica. Tais características são fundamentais para a gestão e controle, objetivadas no presente trabalho, em sistemas de manufatura 4.0.

Segundo Jaskó e colaboradores (2020), o MES é uma solução adequada para vincular o controle no nível do chão de fábrica com as operações de gerenciamento empresarial, fornecendo dados úteis para, por exemplo, as gestões financeira, de relações humanas e de fornecedores. Para eles, na manufatura 4.0, o MES deve ser capaz de lidar em tempo real com o fluxo de dados para transformá-los em informações úteis e direcionadas.

Os sistemas de Gerenciamento das Operações de Manufatura (MOM, do inglês *Manufacturing Operations Management*) geralmente são mencionados como equivalentes ou iguais ao MES. Entretanto, o presente trabalho segue a diferenciação abordada por Filipov e Vasilev (2016), na qual os sistemas MOM estendem a função dos MES. De acordo com os autores, enquanto os MES guiam, rastreiam e registram a manufatura em um local de trabalho (linha ou unidade de processo) – isto é, prioriza funcionalidades como programação e monitoramento da produção e se restringe à manufatura em chão de fábrica –, os sistemas MOM são mais abrangentes ao cobrir todo o processo de manufatura em uma área, o que inclui outras funcionalidades e sistemas, como:

- a) gestão do fluxo de trabalho (digitalização das operações manuais);
- b) gestão da manutenção e os sistemas, incluindo Gestão de Ativos Empresariais (EAM, do inglês *Enterprise Asset Management*) e Sistema Informatizado de Gestão de Manutenção (CMMS, do inglês *Computerized Maintenance Management Systems*);
- c) gestão da qualidade e os sistemas em dois níveis: Sistema de Gestão da Qualidade (QMS, do inglês *Quality Management System*) e Sistema de Gerenciamento de Informações Laboratoriais (LIMS, do inglês *Laboratory*

*Information Management System*);

- d) gestão da logística e o Sistema de Gerenciamento de Armazém (WMS, do inglês *Warehouse Management System*).

Para o presente trabalho será adotado o termo MES, devido ao seu escopo de funcionalidades voltadas à manufatura.

A Associação de Soluções Empresariais de Manufatura (MESA – do inglês *Manufacturing Enterprise Solutions Association*), como uma comunidade de pesquisa e apoio à manufatura, promove discussões para melhoria e padronização de sistemas MES. De acordo com McClellan e Weaver, (2011), ao longo dos anos, a MESA apresentou modelos, que descrevem e apresentam as funcionalidades padrão de um MES. O primeiro modelo, o MESA-11 (lançado em 1996), determinou 11 funcionalidades centrais (MCCLELLAN, WEAVER, 2011):

- a) Alocação e estado de recursos
- b) Programação de operações/detalhes
- c) Despacho de unidades de produção
- d) Controle de documentação
- e) Coleta e aquisição de dados
- f) Gestão do trabalho
- g) Gestão da qualidade
- h) Gerenciamento de processos
- i) Gerenciamento de manutenção
- j) Rastreamento de produtos e genealogia
- k) Análise de desempenho

Segundo os autores, desde o MESA-11 até o modelo atual, foram apresentados outros dois modelos (*Collaborative MES*; *Strategic Initiatives Model*), os quais mantiveram funcionalidades padrão baseadas nestas.

O modelo MESA atual está em desenvolvimento, porém possui um artefato já publicado: *A Framework for Smarter Manufacturing* (em tradução livre, Uma estrutura para uma fabricação mais inteligente). Este visa apresentar de que forma os conceitos tradicionais de manufatura se tornam habilitados para a I4.0 por meio da incorporação de novas tecnologias e da integração das seguintes funções relacionadas (MANUFACTURING ENTERPRISE SOLUTIONS ASSOCIATION, 2022):

- a) Ciclos de vida: Produção; Ativo de Produção; Produto; Cadeia de Suprimentos; Força de trabalho; Ordem-para-Dinheiro.
- b) Ligações entre ciclos de vida: Qualidade; Conformidade; Energia; Análise de Dados; Segurança; Gêmeo / Linha Digital; Modelagem / Simulação.
- c) Tecnologias habilitadoras: IoT Industrial; disponibilidade de *big data*; IA / ML; Realidade Virtual / Aumentada; borda para nuvem; *Blockchain*; Manufatura Aditiva; Robótica; Conectividade sem fio.

De acordo com a descrição e apresentação do artefato, uma organização pode usá-lo para alcançar um objetivo de negócio específico. Isto se dá pelo alinhamento de funções específicas, das três categorias apresentadas acima.

## **2.4 TRABALHOS RELACIONADOS**

Esta subseção apresenta o estado da arte e discute de que formas o presente trabalho pode contribuir para tal. Na primeira parte são mostradas direções de pesquisa atuais. Na segunda parte são discutidos trabalhos que buscam solucionar problemas semelhantes ao deste trabalho.

### **2.4.1 Direções de Pesquisa para Sistemas de Manufatura 4.0**

A literatura científica recente apresenta alguns trabalhos de revisão que analisam o contexto de tecnologias, conceitos e oportunidades e apontam direções de pesquisa para G&C em sistemas de manufatura 4.0. O Quadro 2.2 sintetiza as principais indicações destes trabalhos, que são mais profundamente descritos nos parágrafos a seguir.

Ding e colaboradores (2020) buscaram, em seu trabalho, identificar o estado atual da IA na manufatura. Para os autores, tecnologias de IA, como Aprendizado de Máquina (ML, do inglês *Machine Learning*) e aprendizado profundo, tiveram um desenvolvimento considerável nas últimas décadas e são tendências inevitáveis na manufatura. Estas possibilitam, segundo os autores, novos modos de fabricação que possuem características inteligentes, como autopercepção, autopredição, autoadaptação e auto-otimização. Para enfrentar desafios de tal estágio atual da manufatura, como monitoramento, previsão e otimização de processos, de forma



rápida e escalável, os autores apontam o avanço da capacidade de generalização dos algoritmos de IA industrial. O modelo proposto na seção 3 pretende ter a IA como um de seus conceitos base e, dentro dela, o ML como tecnologia fundamental para possibilitar as habilidades preditivas e adaptativas objetivadas.

Quadro 2.2 Síntese das principais indicações por trabalho

Indicação de pesquisa para G&C em sistemas de manufatura 4.0	Referência
Análise de big data, IA, ML	Ding e colaboradores (2020) Bueno, Godinho Filho e Frank (2020) Usuga Cadavid e colaboradores (2020) Jaskó e colaboradores (2020) Morgan e colaboradores (2021)
IoT	Bueno, Godinho Filho e Frank (2020) Usuga Cadavid e colaboradores (2020)
CPS	Bueno, Godinho Filho e Frank (2020) Morgan e colaboradores (2021)
MES	Jaskó e colaboradores (2020) Lu, Xu e Wang (2020)
SOA	Jaskó e colaboradores (2020) Lu, Xu e Wang (2020) Morgan e colaboradores (2021)
SMA	Jaskó e colaboradores (2020)
OPC UA	Lu, Xu e Wang (2020)

Fonte: Próprio Autor.

Bueno, Godinho Filho e Frank (2020) realizaram uma revisão sistemática para investigar a relação entre o PCP e a I4.0. Eles questionam os recursos inteligentes que a I4.0 pode oferecer ao PCP, as implicações de desempenho nos sistemas de manufatura a partir dessa integração e a influência de fatores ambientais no PCP com capacidades inteligentes. Eles aportam diversas tecnologias, como Análise de *big data*, IA, IoT, CPS, Manufatura Aditiva, Manufatura em Nuvem etc., como novos recursos oferecidos pela I4.0 para controle da produção.

De acordo com Bueno, Godinho Filho e Frank (2020), a flexibilidade é o indicador com o desempenho mais afetado na integração entre o PCP e I4.0. Para eles, apesar

da flexibilidade ser uma vantagem essencial e significativa na I4.0, sua implementação nas empresas é difícil e faltam estudos que abordem este problema. Além disso, os autores afirmam que não encontraram estudos sobre sistemas de suporte à decisão para modelos ou arquiteturas inteligentes de PCP. O modelo proposto no presente trabalho segue os direcionamentos dos autores, principalmente no que se refere às tecnologias indicadas, as quais podem permitir a devida flexibilidade para o controle da produção conforme o requerido pelos modelos de negócio emergentes (em direção à manufatura 4.0).

Usuga Cadavid e colaboradores (2020) exploram a aplicação do ML no PCP na era da I4.0. De acordo com os autores, dentre os casos de uso de aplicação do ML em funções do PCP, os mais abordados na literatura científica recente são o “planejamento e programação inteligente” e o “controle e monitoramento de processos”. No caso do primeiro, segundo os autores, não há abordagens relacionadas à sua intersecção com a “autoexecução de recursos e processos”, que é uma das características da I4.0 apontada por eles. No caso do segundo, ainda segundo os autores, faltam abordagens que atendam a característica de “desenvolvimento de produto centrado no cliente”. Estes fatos, segundo os autores, apontam para uma perspectiva de pesquisa na junção do PCP a recursos físicos auto-otimizáveis a partir da análise de dados relacionados ao cliente.

Usuga Cadavid e colaboradores (2020) concluem que o ML é adequado para geração de conhecimento a partir de dados de PCP, diante da abundância de dados na era da I4.0. Concluem também que os modelos de ML para o PCP efetivamente atribuem capacidades adaptativas e preditivas aos sistemas de manufatura, diante de sua natureza estocástica. Para os autores, O ML aplicado ao PCP ainda é um tópico essencial de pesquisa e apresenta relevantes caminhos, como é o caso da necessidade de reforçar o papel da IoT nesta junção. Para a proposta do presente trabalho, é essencial o conhecimento existente nos dados de PCP, inclusive nos dados relacionados ao cliente, para possibilitar as habilidades adaptativas necessárias ao sistema. Tais habilidades permite que o sistema de manufatura forneça produtos adequados ao desejado pelo cliente (produtos personalizados) no contexto das manufaturas discreta e em batelada.

Jaskó e colaboradores (2020) fazem uma revisão de padrões, metodologias e

ferramentas no desenvolvimento de MES influenciado pela I4.0, identificando tendências atuais e futuras. No que se refere a esse desenvolvimento, o trabalho apresenta uma análise dos requisitos da I4.0, além de uma visão geral sistemática dos métodos e padrões e ainda uma discussão sobre o suporte que modelos semânticos podem oferecer. Os autores enfatizam que os novos MES devem ser desenvolvidos de acordo com condições demonstradas numa análise dos estágios de maturidade da I4.0 (Figura 2.3).

Para o estágio de capacidade preditiva, por exemplo, de acordo com Jaskó e colaboradores (2020), as novas soluções de MES precisam ter funcionalidades de simulação e otimização que corresponderão a necessidade das empresas em prever eventos, indicando também a relação dos MES com o conceito de Gêmeo Digital (GD). Para o estágio de adaptabilidade, segundo os autores, a próxima geração de MES devem ser aptas a suportar a tomada de decisões, direcionadas a vários objetivos, em tempo real. Dentre os papéis do MES no contexto da I4.0 apresentados pelos autores, destacam-se a coordenação das integrações horizontal e vertical, o monitoramento, coordenação, supervisão e programação da manufatura, a gestão dos dados relacionados aos processos de manufatura e, por fim, a implementação e oferta de análises estatísticas complexas e algoritmos avançados de otimização e IA. Para tendências futuras de pesquisa, os autores sugerem mais esforços para encontrar soluções MES com maior descentralização lógica dos recursos computacionais.

Jaskó e colaboradores (2020) ainda evidenciam o potencial da orientação à serviços, na qual, em complemento à descentralização, as entidades são independentes, possuem inteligência e são capazes de tomar decisão. Segundo os autores, neste paradigma, os produtos e materiais inteligentes, que são os CPS, são consumidores de serviços, enquanto os equipamentos e recursos inteligentes, os CPS de Produção (CPPS), são prestadores de serviços. Os autores concluem que a padronização de ferramentas e métodos é a base para soluções de manufatura avançada. Uma outra abordagem apontada pelos autores trata da implementação de sistemas baseados em agentes, os quais permitem integração vertical flexível em plantas distribuídas e proporcionam habilidades adaptativas aos sistemas de manufatura. O MES em conformidade com as características apontadas pelos autores é fundamental no modelo proposto no presente trabalho, pois possui o papel de

orquestrar<sup>5</sup> (junto a uma representação digital orientada a agentes), as atividades de G&C necessárias em sistemas de manufatura 4.0. Entretanto não está no escopo do presente trabalho desenvolver nova solução de MES.

Lu, Xu e Wang (2020) apresentam uma revisão crítica dos padrões para habilitar os processos de produção e a automação dos sistemas de manufatura no contexto da I4.0. Ao apresentar os objetivos e o conceito da manufatura na I4.0 e seu impacto na automação, os autores abordam um conceito simplificado de disponibilidade de ativos físicos como um serviço de manufatura, que são conectados à internet na forma de CPPS, por meio de padrões de comunicação como MTConnect ou a “Comunicações de Plataforma Aberta – Arquitetura Unificada” (OPC UA, do inglês *Open Platform Communications – Unified Architecture*). Para eles, um MES distribuído coordena, monitora e controla a utilização desse ativo, bem como a solicitação deste por outros nós (empresas), em uma interface padronizada, orientada à serviços e baseada em nuvem.

No que se refere à novas iniciativas de padronização, Lu, Xu e Wang (2020) indicam que as atividades de pesquisa sejam direcionadas a superar as barreiras da falta de rastreamento da adoção de padrões, da sobreposição e redundâncias entre padrões e do codesenvolvimento inadequado de padrões com comunidades de usuários. No entanto, os autores acreditam que os padrões existentes forneceram uma base sólida para o desenvolvimento de soluções de automação para a I4.0, de forma que esta precisa fundamentalmente se concentrar nas automações de processos baseada em produtos personalizados e de sistemas de manufatura dotados de inteligência e auto-organizados em rede.

Em concordância com Lu, Xu e Wang (2020), o modelo apresentado na seção 3 se baseia no fato de que a automação atual e seus padrões, em conjunto com novos conceitos e tecnologias, podem possibilitar sistemas de manufatura suficientemente adaptáveis à dinâmica de mercado. O presente trabalho enfatiza a flexibilidade necessária aos sistemas para oferecerem produtos personalizados. A orientação a

---

<sup>5</sup> Entende-se por orquestrar, no presente trabalho, o ato de garantir a execução do papel de cada elemento atuante nas operações da manufatura, de acordo com objetivos de negócio (produtividade, lucro, qualidade, sustentabilidade etc.). Tal orquestração se baseia essencialmente em informações obtidas a partir da reunião de dados e/ou sinais, organizadas em setores de atividades de G&C (funcionalidades).

serviços e os padrões comunicação Máquina a Máquina (M2M, do inglês *Machine to Machine*) citados pelos autores contribui para proporcionar tais flexibilidade e habilidades adaptativas aos sistemas. Estas também são tecnologias utilizadas no modelo apresentado na seção 3.

Morgan e colaboradores (2021) tratam sobre seu conceito proposto, o SRM. Em uma das suas questões de pesquisa eles investigam de que forma deve ser projetado um SRM para habilitar recursos e capacidades ágeis, com menor custo e maior rendimento e para ser intuitivamente operável. Como resposta, os autores recomendam para projetos de SRM a adoção de algoritmos adaptativos e inteligentes de controle, além da tecnologia de controle em tempo real e a implementação de arquiteturas que permitam o controle distribuído e descentralizado. Como potencial oportunidade de pesquisa futura, os autores indicam investigar a maturidade do estado atual de arquiteturas CPS e modelos de máquinas inteligentes. Os direcionamentos dos autores são pertinentes ao presente trabalho, principalmente no que se refere a arquiteturas. O modelo aqui proposto se baseia numa arquitetura que pretende ser modular, distribuída, descentralizada e heterárquica ao aproveitar de oportunidades existentes (como as tecnologias atuais de controle e automação) em conjunto com novos conceitos e tecnologias trazidos pela I4.0.

#### **2.4.2 Modelos e Arquiteturas de Sistemas de Manufatura 4.0**

Relacionado ao estado atual e a perspectivas de pesquisa da manufatura 4.0, um conjunto de trabalhos apresentaram propostas de modelos ou arquiteturas que abrangem conceitos e tecnologias da I4.0 para G&C de sistemas de manufatura. O Quadro 2.3 apresenta uma síntese das principais contribuições e lacunas destes trabalhos, que são descritos mais profundamente nos parágrafos seguintes. Tais trabalhos são considerados aqui como o estado da arte de modelos – e de tecnologias e conceitos que as compõe – semelhantes ao proposto na seção 3. Em geral, o presente trabalho pretende aproveitar de abordagens pertinentes e preencher lacunas deixadas por tais propostas.

Zhuang, Liu e Xiong (2018) propuseram uma estrutura para G&C inteligente da produção, baseado em GD, para chão de fábrica de montagem de produtos complexos (em termos de sua composição, tecnologia, processo de fabricação,

gerenciamento de seu projeto e demanda do cliente). O caso utilizado para a análise na pesquisa foi a montagem de satélites. Segundo os autores, os componentes da estrutura são o chão de fábrica, o GD do chão de fábrica, a plataforma de armazenamento e gerenciamento de *big data* e a plataforma de serviço e aplicação.

Quadro 2.3 Síntese das principais abordagens e lacunas encontradas, por trabalho

Referência	Abordagem utilizada na arquitetura ou modelo	Lacuna ou oportunidade de evolução
Zhuang, Liu e Xiong (2018)	<ul style="list-style-type: none"> <li>• Representação digital de dispositivos físicos (por GD);</li> <li>• Análise de big data, IA, ML;</li> <li>• SOA</li> </ul>	<ul style="list-style-type: none"> <li>• MES</li> </ul>
Arm e colaboradores (2021)	<ul style="list-style-type: none"> <li>• Representação digital de dispositivos físicos;</li> <li>• Controle distribuído e descentralizado</li> </ul>	<ul style="list-style-type: none"> <li>• SMA;</li> <li>• SOA</li> </ul>
Kruger e colaboradores (2011)	<ul style="list-style-type: none"> <li>• SMA;</li> <li>• SOA;</li> <li>• Análise de big data, IA, ML;</li> <li>• setorização de atividades de G&amp;C (funcionalidades)</li> </ul>	<ul style="list-style-type: none"> <li>• Modelo genérico;</li> <li>• MES;</li> <li>• Arquitetura distribuída, descentralizada e heterárquica;</li> <li>• SMA</li> </ul>
Tang e colaboradores (2017)	<ul style="list-style-type: none"> <li>• SMA;</li> <li>• MES;</li> <li>• Análise de big data, IA, ML;</li> <li>• M2M;</li> <li>• OPC UA</li> </ul>	<ul style="list-style-type: none"> <li>• SOA;</li> <li>• Arquitetura distribuída, descentralizada e heterárquica</li> </ul>
Kim e colaboradores (2020)	<ul style="list-style-type: none"> <li>• Análise de big data, IA, ML;</li> <li>• SMA</li> </ul>	<ul style="list-style-type: none"> <li>• Modelo genérico;</li> <li>• MES</li> </ul>
Wan e colaboradores (2020)	<ul style="list-style-type: none"> <li>• Análise de big data, IA, ML;</li> <li>• SMA;</li> <li>• SOA</li> </ul>	<ul style="list-style-type: none"> <li>• Modelo genérico</li> </ul>
Buckhorst, Grahn e Schmitt (2022)	<ul style="list-style-type: none"> <li>• SMA;</li> <li>• Controle descentralizado;</li> <li>• SOA;</li> <li>• OPC UA;</li> <li>• Análise de big data, IA, ML;</li> </ul>	<ul style="list-style-type: none"> <li>• MES</li> </ul>
Shojaeinasab e colaboradores (2022)	<ul style="list-style-type: none"> <li>• MES;</li> <li>• Análise de big data, IA, ML;</li> <li>• M2M;</li> <li>• OPC UA;</li> <li>• SMA</li> </ul>	<ul style="list-style-type: none"> <li>• Arquitetura distribuída, descentralizada e heterárquica</li> </ul>

Fonte: Próprio Autor.

O chão de fábrica de montagem física, segundo Zhuang, Liu e Xiong (2018), trata dos dispositivos físicos. O GD do chão de fábrica, para os autores, é a reconstrução e o mapeamento digital do chão de fábrica no espaço virtual. Segundo os autores, o chão de fábrica e o GD trocam dados, informações ou conhecimentos, por meio da plataforma de armazenamento e gerenciamento de *big data*. Esta, de acordo com os autores, atua como uma única fonte de dados na estrutura e fornece suporte de dados para os outros três componentes. A plataforma de serviço e aplicação refere-se, segundo os autores, ao conjunto de tecnologias que suportam os requisitos funcionais e de destino da G&C inteligente da produção.

Ainda segundo Zhuang, Liu e Xiong (2018), tal plataforma realiza o monitoramento, a previsão e o controle de otimização de recursos de fabricação, atividades de produção e processos de produção. É composta também, segundo os autores, por dois subcomponentes: uma plataforma de serviços de previsão e uma plataforma de serviços de G&C da produção. De acordo com os autores, a plataforma de serviço de previsão contém funcionalidades como previsão de qualidade do produto, previsão de progresso de produção e previsão de gargalo de produção, enquanto a plataforma de serviço de G&C da produção compreende funcionalidades como montagem otimização de processos, otimização logística de produção e otimização de recursos de fabricação.

A estrutura proposta por Zhuang, Liu e Xiong (2018) é interessante para ser aproveitada, em partes, no modelo aqui proposto. Os fatos desta ser orientada a serviços (plataforma de serviço e aplicação) e de utilizar de dados para geração de conhecimento útil para G&C, além de abordar um caso de uso de manufatura discreta, são determinantes. Entretanto pretende-se que o GD proposto no modelo aqui apresentado atue, se utilizado, como representação digital inteligente do processo de manufatura. Está fora do escopo do modelo aqui proposto a reconstrução e mapeamento do chão de fábrica em caráter geográfico, espacial ou estético.

Arm e colaboradores (2021) apresentam um método de gestão inteligente da produção – *Asset Administration Shells* (AAS) – que utiliza ativos inteligentes. Segundo os autores, o AAS representa virtualmente, digitalmente e ativamente um componente de produção, num sistema de produção de I4.0, ao estabelecer uma interface entre as variantes de produção física e virtual. Para eles, cada ativo virtual,

como por exemplo, um produto, uma máquina, um robô ou um transportador, possui seu encapsulamento de administração. Segundo os autores, esses AAS têm capacidade de se comunicar e negociar prioridades e requisitos de produção de acordo com um conjunto de regras pré-especificadas, o que possibilita a incorporação de recursos inteligentes nas operações de manufatura.

O conceito proposto por Arm e colaboradores (2021) é originado de uma nova interpretação da gestão, na qual os componentes relevantes são integrados tanto horizontalmente quanto verticalmente. De acordo com eles, esse conceito muda a arquitetura do sistema de controle de processo industrial para uma forma distribuída e descentralizada, atribuindo flexibilidade na programação de operações, respostas a falhas e personalização do produto. O presente trabalho aproveita do conceito proposto, e suas características, pelos autores, porém em uma abordagem diferente. Pretende-se estabelecer uma estrutura mais distribuída e descentralizada, orientada a serviços e a agentes como representantes digitais inteligentes de componentes de produção, em lugar do AAS.

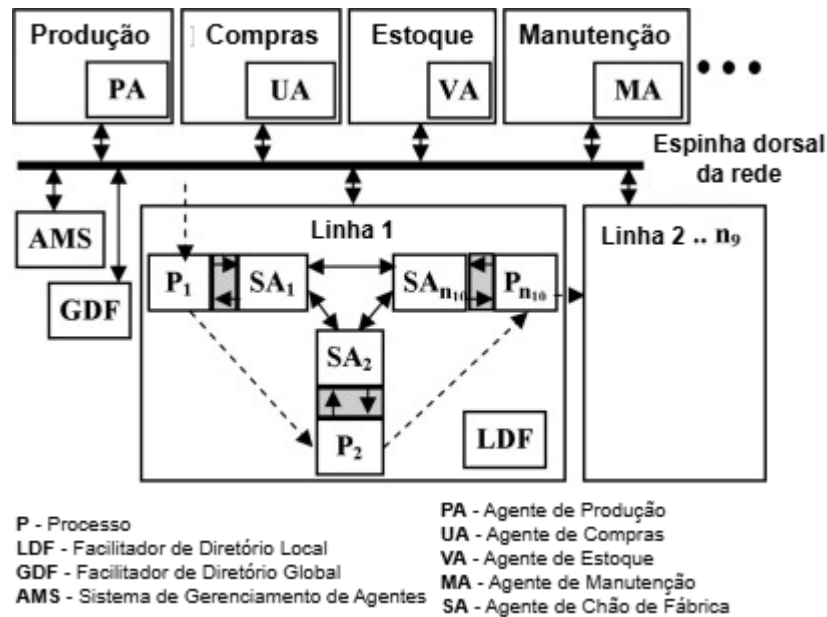
Kruger e colaboradores (2011) ressaltam a importância de arquiteturas formais para soluções inteligentes, heterogêneas e interoperáveis, que agilizem as implementações e a melhoria contínua do controle de processos de manufatura. Em sua arquitetura orientada a agentes (Figura 2.4), os autores classificaram seus agentes por níveis departamentais e descrevem sua integração:

- a) agente de planejamento e produção: estabelece a meta para cada processo ou linha de produção;
- b) agente de compras: é responsável por garantir que os materiais, componentes e equipamentos necessários sejam solicitados nas quantidades corretas e cheguem no prazo;
- c) agente de estoque: é responsável por manter todas as ferramentas, peças sobressalentes, matérias-primas e equipamentos necessários para atender ao processo de fabricação, além de fornecer um *buffer* caso ocorram problemas durante o processo de compra;
- d) agente de manutenção: é responsável pelo atendimento reativo, proativo e preditivo de eventuais falhas que ocorram nos processos;
- e) agentes de chão de fábrica: são responsáveis por garantir que os objetivos do



“agente de planejamento e produção” sejam executados, o que inclui informar o “agente de manutenção” sobre o estado da máquina, bem como informar o “agente de estoque” sobre as necessidades atuais e futuras, além de negociar com outros “agentes de chão de fábrica” para encontrar o equilíbrio ideal entre os processos locais e globais.

Figura 2.4 Integração em toda a empresa da arquitetura de agente de software



Fonte: Adaptado de Kruger e colaboradores (2011).

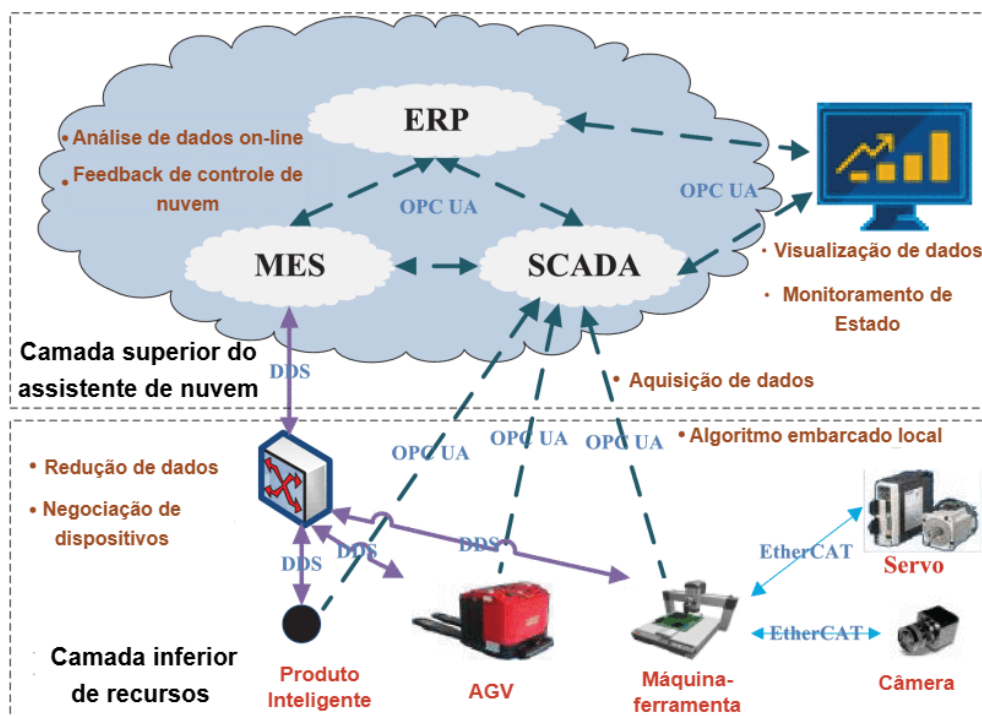
De acordo com os autores, a arquitetura é modular e visa maximizar a instanciação e integração de vários algoritmos que atuam para estabelecer trocas ideais (otimizadas) entre vários objetivos técnicos e econômicos de manufatura.

O trabalho de Kruger e colaboradores (2011) foi um dos primeiros a propor uma arquitetura orientada a agentes e um modelo de setorização das atividades de G&C semelhante às funcionalidades MES. A arquitetura proposta pode ser considerada como um importante alicerce para novas propostas de arquiteturas orientadas à agentes e à serviços. Entretanto, a arquitetura tem um foco maior para implementação na usinagem. Espera-se um modelo genérico aplicável em diferentes indústrias. Além disso não foi objetivo da pesquisa detalhar componentes físicos na arquitetura, como também integrar o MES aos agentes e alcançar maiores níveis de descentralização, distribuição e heterarquia. Segundo os próprios autores, a proposta ainda pode ser

evoluída no que diz respeito à Sistemas Multiagentes (SMA). O modelo proposto no presente trabalho pretende contemplar esses fins.

Tang e colaboradores (2017) propuseram uma arquitetura distribuída, baseada em agentes e assistida em nuvem, para sistemas de manufatura (Figura 2.5). A arquitetura compreende duas camadas: a camada inferior de recursos e a camada superior do assistente de nuvem. Segundo os autores, a camada de recursos, na qual são representados os dispositivos, viabiliza a IoT industrial no sistema de manufatura, além de permitir que objetos inteligentes colaborem uns com os outros para concluir tarefas de produção sem um agendador centralizado. A camada do assistente de nuvem, de acordo com os autores, coleta dados da camada de recursos e estabelece o plano de agendamento ideal global, evitando ótimos locais possíveis no agendamento distribuído, por meio da análise de dados. Segundo eles, esses planos então são devolvidos como sugestões de agendamento para as plantas.

Figura 2.5 Arquitetura auto-organizada assistida por nuvem



Fonte: Adaptado de Tang e colaboradores (2017).

Ainda de acordo com Tang e colaboradores (2017), na camada do assistente de nuvem, o SCADA é estendido para capacidades maiores de serviço e gerenciamento

de dados e combina-se com o MES, que é estendido para maiores capacidades de comunicação com a planta. Segundo eles, esses dois sistemas cooperam com o ERP para adquirir funções como agendamento de recursos, gerenciamento de tarefas, processamento de dados e visualização.

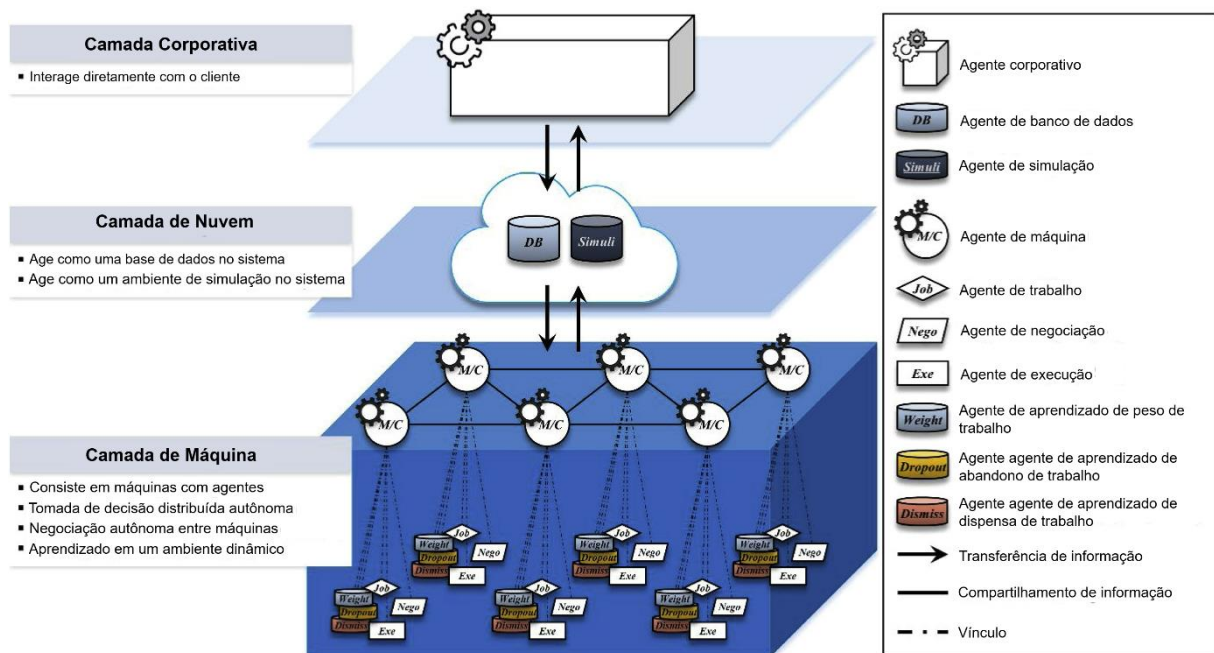
Tang e colaboradores (2017) explicam que os recursos da camada inferior competem e cooperam em um contexto de fabricação descentralizado com capacidade de auto-organização, além de contribuírem com vantagens complementares para obter flexibilidade no processamento de produtos de vários tipos. Dessa forma, segundo eles, as entidades inteligentes na arquitetura podem ser modeladas como SMA para construir um sistema de manufatura inteligente.

As entidades inteligentes podem ser classificadas, segundo Tang e colaboradores (2017), em quatro tipos de agentes, de acordo com suas funções e responsabilidades: agentes de sugestão, que representam o componente de *software* na nuvem responsável por processar pedidos e gerar sugestões de agendamento; agentes de produto, que representam os produtos que precisam ser processados e agem como nós autônomos e colaborativos para serem capazes de interagir com outros agentes; agentes de usinagem, que podem realizar usinagem e armazenamento de operações, referentes a máquinas-ferramentas, equipamentos de teste, entre outros; e agentes de transporte, que podem transportar o agente do produto da localização atual até o destino, referindo-se a esteiras transportadoras, veículos autoguiados, entre outros.

Para oferecer interoperabilidade e comunicação em tempo real, Tang e colaboradores (2017) apresentam dois métodos na arquitetura: OPC UA para comunicação máquina-nuvem e Serviço de Distribuição de Dados (DDS, do inglês *Data Distribution Service*) para comunicação M2M. Apesar de sua robustez em apresentar propriedades e tecnologias que permitem a atribuição de habilidades adaptativas aos sistemas de manufatura, há oportunidade de evoluir a arquitetura de Tang e colaboradores (2017) para maiores níveis de descentralização e heterarquia, de forma a possibilitar sistemas de manufatura a alcançar maiores estágios de maturidade da I4.0. O modelo proposto a seguir pretende atender essa oportunidade com uma arquitetura heterárquica e orientada a serviços, além de utilizar elementos apresentados, como SMA, MES e M2M.

Kim e colaboradores (2020) implementaram SMA e Aprendizado por Reforço (AR) como bases para a sua proposta de um sistema de manufatura inteligente, flexível e ágil. Segundo os autores, a arquitetura (Figura 2.6) objetiva oferecer ao sistema as capacidades de tomada de decisões autônomas por máquinas, interações entre máquinas, além das capacidades de sistema tomar decisões finais e de aprender. O sistema consiste em três camadas: a corporativa, a de nuvem e a de máquina. De acordo com os autores, a classe de agente corporativo, na camada corporativa, interage diretamente com o cliente e é responsável por receber os pedidos e entregar ao cliente informações do cronograma de trabalho. Na camada de nuvem existem os agentes de banco de dados e de simulação. O agente de banco de dados, segundo os autores, armazena as informações do pedido do cliente e o plano e cronograma de produção referente ao pedido, enquanto o agente de simulação fornece um ambiente para os agentes responsáveis pelo AR na camada de máquina.

Figura 2.6 Arquitetura de sistema de manufatura inteligente



Fonte: Adaptado de Kim e colaboradores (2020).

A camada de máquina, de acordo com Kim e colaboradores (2020), é responsável pela tomada de decisão e negociação autônoma entre os agentes de máquina para decidir a quantidade de produção adequada e atribuir tarefas às

máquinas. Nesta camada contém seis classes de agentes: agente de trabalho, agente de negociação, agente de aprendizado de peso de trabalho, agente de aprendizado de abandono de trabalho, agente de aprendizado de dispensa de trabalho e agente de execução. Segundo os autores, estes agentes fornecem às máquinas autonomia, sociabilidade e capacidade de aprendizado, o que possibilita os agentes a distribuir autonomamente novos trabalhos por meio de uma consciência do ambiente e de uma comunicação mútua, além de terem capacidade de aprendizado para melhorar a tomada de decisões.

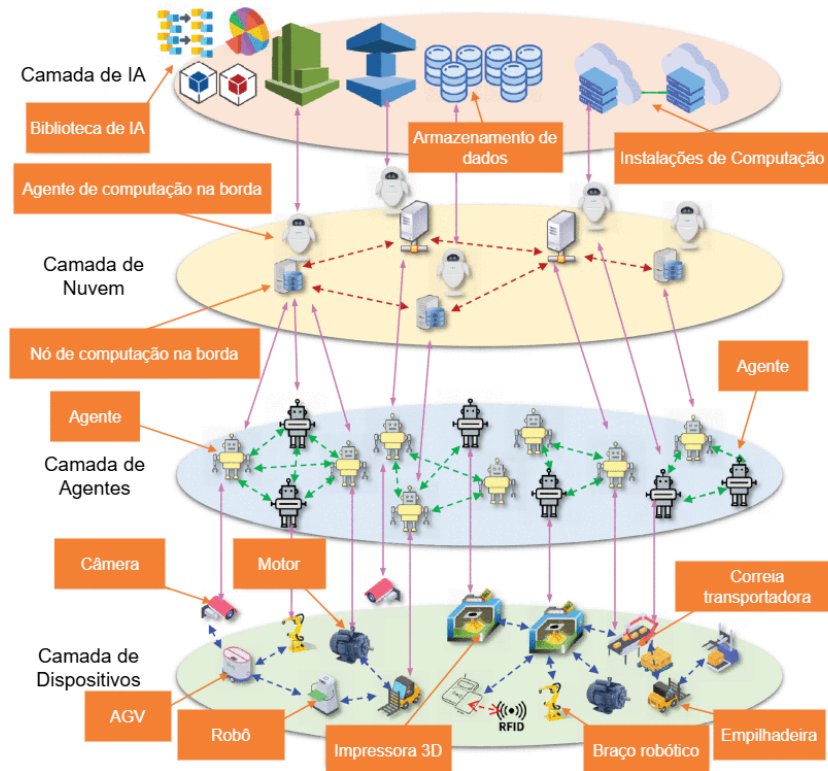
O estudo de Kim e colaboradores (2020) apresenta importantes contribuições no que diz respeito à tomada de decisão distribuída autônoma de máquinas num sistema de manufatura por meio da implementação de SMA, bem como ao método para programar a produção, além da capacidade do sistema de se adaptar e tomar decisões apropriadas em ambientes que mudam dinamicamente, por meio da implementação de AR. Além de alcançar habilidades de aprendizado, cooperação e tomada de decisão, por meio de SMA, como a proposta de Kim e colaboradores (2020), o modelo proposto na seção 3 busca alcançar uma arquitetura formal mais genérica, que incorpore e descreva a relação e a comunicação entre seus componentes, e que permita a aplicação direta de atividades de G&C a partir das funcionalidades MES.

Wan e colaboradores (2020) abordam dispositivos, interações e serviços inteligentes para propor uma arquitetura orientada por IA para manufatura personalizada (Figura 2.7). Os dispositivos inteligentes (robôs, transportadores e outras plataformas controladas básicas) compõem a camada física da arquitetura, a qual deve atender a requisitos de tempo real. Para isso os autores indicam implementação de servidores de computação em borda e algoritmos de ML em dispositivos de baixa potência.

A conexão da camada de dispositivos às camadas de IA e de serviços, na arquitetura de Wan e colaboradores (2020), é papel da interação inteligente, segundo os autores. Esta é composta por dois módulos vitais: o de dispositivos básicos de rede (pontos de acesso, *switches*, roteadores etc.), que é estático, e o módulo composto pelos elementos dinâmicos (protocolos de rede e comunicações, interação de informações etc.). De acordo com os autores, a camada de IA contém algoritmos

executados em diferentes plataformas de computação, como servidores de borda ou em nuvem. Segundo eles, o componente da arquitetura denominado Serviços de Manufatura Inteligente inclui visualização de dados, manutenção do sistema, previsões e análise de mercado.

Figura 2.7 Dispositivos de manufatura assistidos por computação de borda



Fonte: Adaptado de Wan e colaboradores (2020).

Para alcançar controle flexível e rápido sobre dispositivos de manufatura em quantidade, e conseqüentemente a rápida reestruturação e reutilização para pequenos lotes de produtos personalizados, Wan e colaboradores (2020) propõem o paradigma de SMA. A arquitetura proposta é genérica e abrangente, além de convergir para o que a comunidade científica e industrial tem solicitado quanto às tecnologias da I4.0 e necessidades de G&C na manufatura. No entanto, pode evoluir para um modelo mais focado na aplicação imediata, com maior capacidade de “conecte e produza”. Esta deve incluir, além de especificações de componentes, um modelo descritivo de implementação que permita um sistema de manufatura atingir níveis mais avançados no modelo de maturidade da I4.0. O modelo proposto adiante

pretende preencher tal oportunidade.

Buckhorst, Grahn e Schmitt (2022) propuseram uma arquitetura de G&C baseada nas vantagens inerentes do paradigma recentemente apresentado Montagem Móvel Sem Linha (LMAS, do inglês *Line-less Mobile Assembly*). De acordo com os autores, LMAS é um paradigma organizacional, para sistemas de montagem, que oferece flexibilidade, reconfigurabilidade e adaptabilidade em nível de fábrica, estação e recurso ao explorar a mobilidade tecnológica da combinação entre robôs industriais clássicos e veículos autonomamente guiados (AGV, do inglês *Automated Guided Vehicle*). Os autores afirmam que a arquitetura proposta visa controle descentralizado e gestão das operações, baseado em Sistemas Holônicos de Manufatura (SHM) e SMA, específico para LMAS.

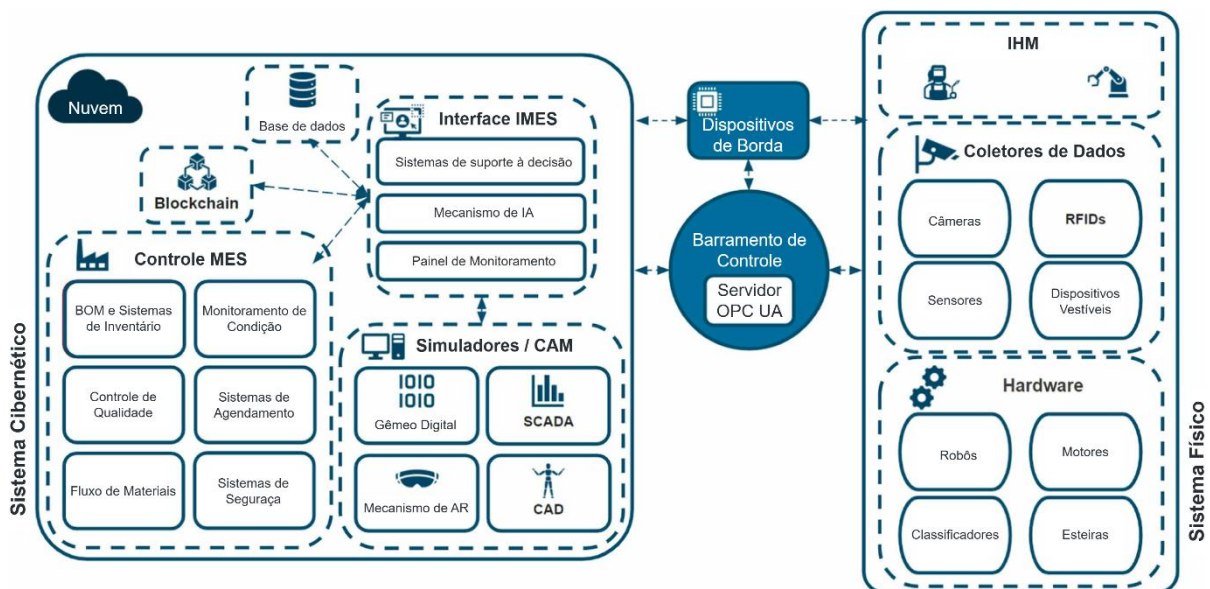
Segundo Buckhorst, Grahn e Schmitt (2022), na interface de comunicação da arquitetura, as entidades que fazem parte de uma mesma rede fabril podem trocar mensagens, oferecer ou solicitar serviços. As tecnologias de comunicação indicadas como relevantes pelos autores podem ser, OPC UA, o Transporte de Telemetria de Enfileiramento de Mensagens (MQTT, do inglês *Message Queuing Telemetry Transport*), o Transferência de Estado Representacional (REST do inglês, *Representational State Transfer*) ou outras em SOA. Para os autores, o agente hólón é um nó de *software* autônomo que é responsável por atingir seus objetivos transitórios por meio de tomada de decisão. De acordo com eles, o “modelo de decisão” de um agente o faz interpretar e interagir com seu ambiente, para fins de percepção, otimização local e interação, além de ser capaz de evoluir (por meio de ML, por exemplo) ou ser atualizado. Os autores afirmam que a arquitetura permite obter o grau desejado de flexibilidade nas tarefas de MES. O modelo proposto no presente trabalho se apoia em parte da abordagem de Buckhorst, Grahn e Schmitt (2022), incorporando SMA e tecnologias de comunicação em SOA, e a escala para implementação em sistemas de manufatura discreta e em batelada.

Shojaeinasab e colaboradores (2022) além de realizarem uma revisão sistemática acerca de MES inteligentes, propõem um modelo conceitual para realização de tarefas de MES com capacidade de previsão e alta adaptabilidade (Figura 2.8). Eles afirmam que, para tomar decisões de controle em tempo real, o MES requer o recebimento de atualizações de estado de todos os dispositivos conectados



no chão de fábrica em tempo real, além de enviar atualizações e instruções a esses dispositivos para o gerenciamento da produção. Para isso, eles enfatizam que a comunicação M2M entre dispositivos, *software* e bancos de dados é uma componente chave, além de habilitar arquiteturas de fábricas inteligentes, especialmente no nível do MES. De acordo com os autores, o OPC UA é o protocolo de comunicação industrial padronizado mais amplamente adotado atualmente e apresenta rapidez e segurança para exercer esse papel. Eles destacam também a importância de dispositivos de borda nos sistemas de manufatura inteligentemente orquestrados por MES, os quais permitem que aplicativos de IA processem dados onde são produzidos.

Figura 2.8 Componentes do modelo conceitual IMES (MES Inteligente)



Fonte: Adaptado de Shojaeinasab e colaboradores (2022).

Um outro componente proposto por Shojaeinasab e colaboradores (2022) é uma interface única e integrada de gestão, que reúne os diversos elementos distintos que compõem um MES. Para eles, esta permite não apenas a gestão baseada no usuário, mas também inclui Interfaces de Programação de Aplicações (API, do inglês *Application Programming Interface*) e protocolos padronizados para integração com outros módulos de *software*. Em relação ao controle do MES sobre as operações de chão de fábrica, a proposta dos autores é o uso de IA para automação e aumento da eficiência da linha de produção. Quanto à gestão dos dados, é recomendado pelos



autores o armazenamento de *logs* do sistema e *logs* de produtos, que consistem em seu estado e recursos, em um sistema de banco de dados adequado, além de uma versão redundante dos dados em um sistema *blockchain*, o qual também suportará a rastreabilidade da cadeia de suprimentos.

De acordo com Shojaeinasab e colaboradores (2022), a conectividade IoT e o emprego de controladores no sistema de manufatura possibilitam o MES a se comunicar e controlar dispositivos de campo (máquinas e *hardware* para operações de chão de fábrica). No que se refere à coleta de dados, para os autores, a nova geração MES precisa coletar de forma automática e eficaz os dados do chão de fábrica usando sensores, Identificação por Radiofrequência (RFID, do inglês *Radio-Frequency Identification*), diferentes tipos de câmeras e dispositivos vestíveis.

A proposta de Shojaeinasab e colaboradores (2022) é abrangente e complexa em relação ao presente trabalho. Não está incluso no escopo do modelo aqui proposto fatores como melhorar a integrabilidade intrínseca do MES com outros sistemas e a gestão dos dados (utilizando *blockchain*). Entretanto, o modelo proposto a seguir pretende ser capaz de operar em sistemas tão avançados quanto os abordados pelos autores. Além disso, o modelo também é coerente com a abordagem dos autores ao utilizar o conceito de SMA para estabelecer agentes, como partes digitais e inteligentes de dispositivos físicos, para comunicarem e cooperarem com o objetivo de tomar decisões de G&C da manufatura, baseado em dados e informações.

*Neste capítulo explica-se o modelo proposto bem como as tecnologias e métodos utilizados para a sua criação.*

## **MODELO PARA GESTÃO E CONTROLE INTELIGENTES DA MANUFATURA**

O modelo aqui proposto, denominado SAMOM (do inglês *Smart Adaptive Manufacturing Operations Management*), projeta uma arquitetura para operação de sistemas de manufatura com habilidades inteligentes e adaptativas. Ele tem como objetivo proporcionar e compatibilizar G&C em sistemas de manufatura 4.0, ou em sistemas que buscam evoluir para operações de manufatura 4.0. O foco do modelo SAMOM é em sistemas que operam sob processos de manufatura discretos ou em batelada, que possuem complexidades inerentes à demanda por variedade de processos e a personalização de produtos. O escopo do modelo foca nas operações de produção, contudo, dadas as interrelações, pode ser escalado para manutenção, qualidade, logística e cadeias de suprimentos e de distribuição.

O modelo SAMOM define a estrutura e as regras básicas de interação entre dispositivos físicos utilizados na manufatura, respectivas representações digitais destes dispositivos e atividades de G&C, estas últimas enquanto funcionalidades de sistemas de gerenciamento das operações de manufatura. Esta estrutura e suas regras visam à atuação dos elementos do sistema de forma distribuída, heterárquica e orientada a serviços.

O modelo também trata da atribuição de inteligência ao sistema de manufatura, a partir do uso de algoritmos lógicos combinados com o conhecimento originado em dados historiados. Para isso, um sistema de manufatura apto a receber uma instância do modelo SAMOM deve estar classificado ao menos no estágio 3 de maturidade tecnológica da Figura 2.3. Em outras palavras, este sistema deve estar conectado de tal forma que os processos podem ser monitorados. Neste caso, o modelo SAMOM pode o estabelecer no estágio 4, devido ao requisito de se armazenar dados dos processos ao longo do tempo, que por sua vez ajudam no diagnóstico de eventos. A partir disso, o modelo SAMOM pode então introduzir o sistema no estágio 5, de forma que este passa a possuir habilidades preditivas baseado nos dados armazenados.

O modelo SAMOM demonstra capacidade de operar em sistemas estabelecidos no estágio 6 de maturidade da Figura 2.3. Entretanto, não é o objetivo do modelo tal plenitude, mas sim atribuir a referida inteligência de forma que sejam alcançadas habilidades adaptativas. Ressalta-se que não está no escopo do presente trabalho a verificação e a classificação de maturidade tecnológica de sistemas.

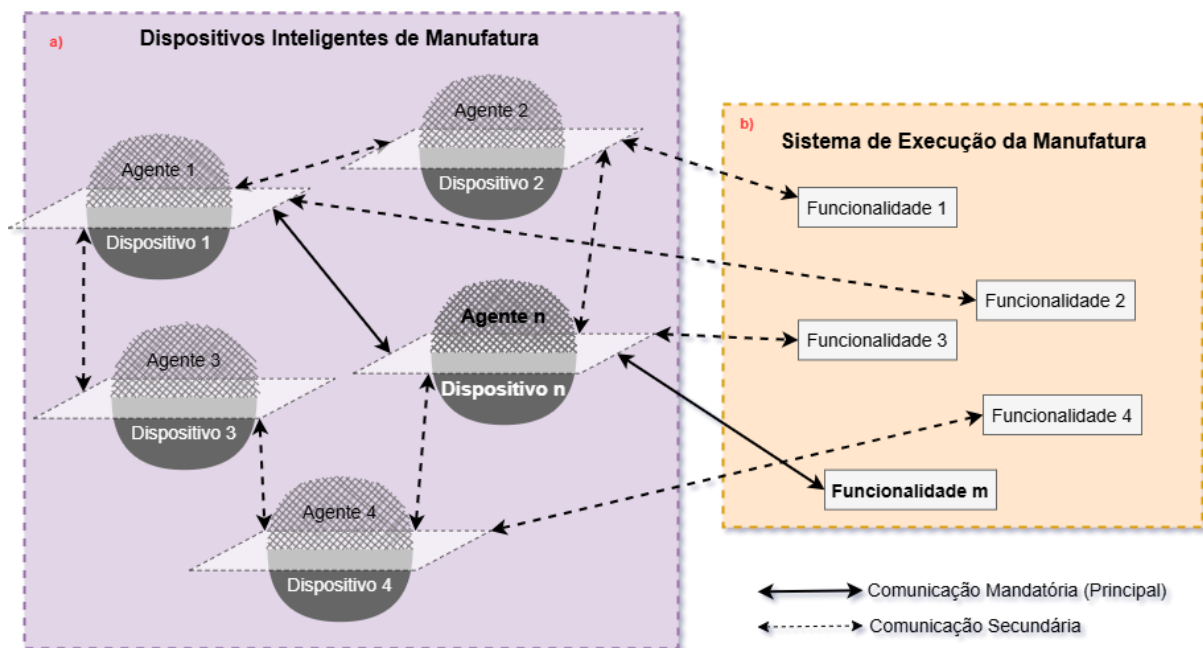
### **3.1 ARQUITETURA PRINCIPAL**

A Figura 3.1 apresenta a arquitetura principal do modelo SAMOM, a qual é orientada a serviços (SOA) e proporciona uma visão geral e genérica do modelo. Esta é composta por dois sistemas: o de Dispositivos Inteligentes de Manufatura (DIM) e o MES. O sistema de DIM, na Figura 3.1 a), é composto por entidades que atuam nos processos de manufatura como, máquinas, operadores, equipamentos, produtos, robôs etc. Tais dispositivos inteligentes são formados por uma parte física, os dispositivos físicos, e suas respectivas representações digitais, os agentes. O MES, na Figura 3.1 b), reúne funcionalidades de G&C da manufatura.

O modelo SAMOM propõe comunicação entre os DIM e as funcionalidades MES, de forma que os primeiros, que podem ser computacionalmente inteligentes, solicitam serviços e enviam, bem como recebem, dados e informações ao MES. Há comunicação também entre os próprios DIM, o que possibilita cooperação entre eles em sua operação nos sistemas de manufatura. As setas de comunicação mandatória indicam que é imprescindível que, na operação dos sistemas sob o modelo proposto,

haja comunicação de DIM entre si e com funcionalidades MES (isto é, se existir ao menos uma de tais comunicações, o modelo SAMOM está estabelecido). As setas de comunicação secundárias indicam, em ambos os sentidos (leitura/recebimento e escrita/envio), que nem todos os DIM obrigatoriamente comunicam entre si ou com as funcionalidades MES, mas apenas de acordo com a conveniência e pertinência específicas de cada implementação. Contudo, é recomendável que, nas aplicações do modelo SAMOM, os DIM estejam aptos a comunicarem entre si e com o MES, pelo fato de que a arquitetura se propõe a ser modular e escalável. A existência de tais comunicações, em ambos os sentidos, demonstra a capacidade de execução de atividades de G&C (funcionalidades MES) por cada DIM de forma autônoma. Demonstra também uma capacidade de inserção de novas execuções de atividades de G&C (interação entre DIM e MES) sem necessariamente interferir nas já existentes.

Figura 3.1 Arquitetura Principal do SAMOM



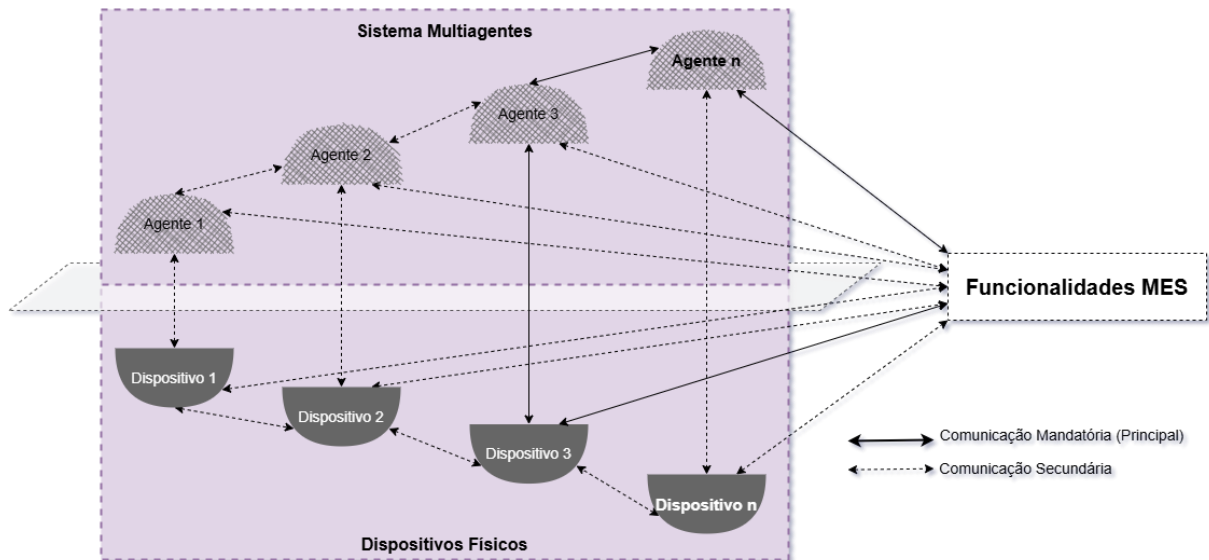
Fonte: Próprio Autor.

### 3.2 DISPOSITIVOS INTELIGENTES DE MANUFATURA

Na arquitetura dos DIM, apresentada na Figura 3.2, há duas camadas: a do SMA e a de dispositivos físicos. Esta arquitetura demonstra a composição dos DIM em suas

partes digital e física, o que inclui a comunicação entre estas partes, entre os componentes na mesma camada e de cada componente com as funcionalidades MES.

Figura 3.2 Arquitetura dos DIM



Fonte: Próprio Autor.

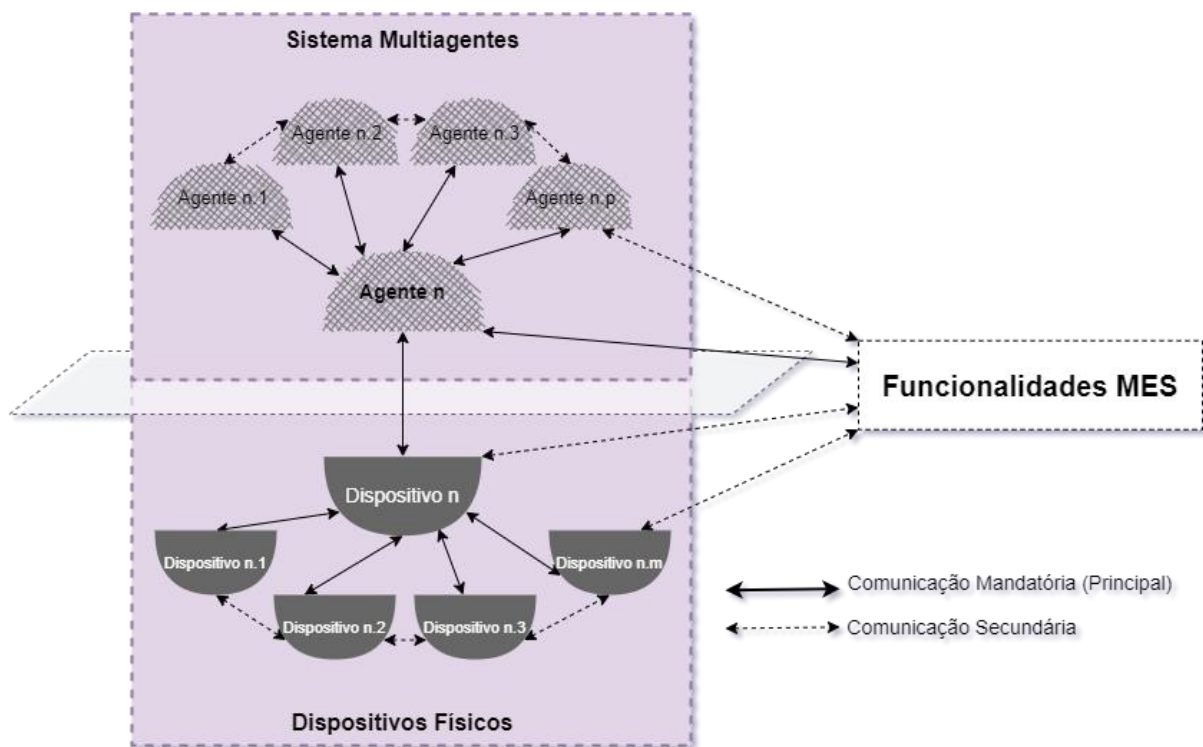
A representação digital em tal composição complementa seu respectivo DF, mas não o copia. Não é objetivo e escopo do modelo SAMOM estabelecer GD de sistemas de manufatura. O principal motivo é a não obrigatoriedade do modelo SAMOM de estabelecer interação síncrona entre os dispositivos e suas representações digitais. Outro motivo é que não faz parte do escopo do presente trabalho o estabelecimento de representações digitais visuais. Entretanto, o modelo SAMOM estabelece uma estrutura que pode ser usada como base de um GD, ou cooperar com este.

A representação das comunicações é análoga à descrita anteriormente para a arquitetura principal: as setas de comunicação mandatória indicam condição para caracterizar o modelo proposto e as setas de comunicação secundárias indicam a não obrigatoriedade. Entretanto, para estas últimas, é recomendada a capacidade de comunicação em todos os casos e a existência sempre que pertinente.

Os dispositivos físicos, na parte inferior das Figuras 3.2 e 3.3, representam componentes que fazem parte fisicamente de processos de manufatura, como

atuadores, sensores, controladores, equipamentos, ferramentas, robôs, produtos etc. Cada dispositivo físico, ou conjunto destes dispositivos, se comunica com seu respectivo agente, ou conjunto destes agentes. Dessa forma, a parte física se une a parte digital para compor os DIM. Um agente, no escopo do presente trabalho, deve ser entendido como um elemento computacional (*software*) com capacidade de percepção (recebimento de sinais e dados), processamento (cálculos, lógicas, inferências etc.) e ação (envio de novos sinais e dados) sobre um ambiente físico e/ou virtual.

Figura 3.3 Conjuntos de agentes e dispositivos físicos para um DIM genérico



Fonte: Próprio Autor.

Conforme o padrão de comunicação, mandatória e secundária, é possível, mas não obrigatório, que haja comunicação entre alguns dispositivos físicos. No caso do exemplo genérico na Figura 3.3, os dispositivos de 1 a m se comunicam e fazem parte de um conjunto (Dispositivo n) que possui seu respectivo agente. Em outros casos, um só dispositivo físico pode ter seu respectivo agente. Um conjunto contendo um sensor, um atuador e um controlador, por exemplo, formam um módulo de manufatura

que pode ter um único agente correspondente. De outro modo, um único atuador, de um tipo específico, pode ter seu agente correspondente.

Os agentes (camada Sistema Multiagentes) podem ser dotados de inteligência, por meio de lógicas e técnicas de IA, principalmente modelos de ML, estatística e otimização matemática. Baseados em dados e informações, estes agentes utilizam seus atributos para realizar inferências, predições e adaptações sobre os dispositivos físicos de acordo com as necessidades de G&C na dinamicidade dos processos de manufatura. Também conforme o padrão de comunicação mandatória e secundária, deve haver comunicação entre agentes, para que sejam possibilitadas coordenação e cooperação entre estes, a fim de serem tomadas as melhores decisões possíveis pelos agentes, o que o caracteriza como um SMA. No caso do exemplo genérico da Figura 3.3, os agentes de 1 a  $p$  se comunicam e fazem parte de um conjunto de agentes (Agente  $n$ ), que possui seu respectivo dispositivo físico. Em outras situações, um único agente pode ter seu respectivo único dispositivo físico. É possível também que dois agentes possuam diferentes papéis em um conjunto que representa a parte digital de uma máquina, por exemplo (isto é, um dispositivo físico pode possuir mais de um agente). Também conforme o padrão de comunicação, mandatória e secundária, não é obrigatório que todos os dispositivos físicos se comuniquem diretamente com seu agente, como mostra a Figura 3.2.

Os agentes são modelados, desenvolvidos e operam em um ambiente computacional, por meio de linguagens de programação e *software* apropriados como, por exemplo, Simulink/Matlab, RapidMiner, Python ou Java. Este ambiente computacional pode ser uma rede local ou um único computador, além de também poder ser explorada a implementação em nuvem, a depender do caso.

Determinados tipos de dispositivos físicos podem ter intrinsecamente sua representação digital e até mesmo inteligência. Nestes casos, os agentes podem complementar as capacidades já existentes como também adequar os dispositivos aos padrões do modelo proposto. Um exemplo comum são alguns tipos de robôs (como robôs com sistemas de visão computacional integrados e robôs colaborativos), que podem possuir determinadas capacidades preditiva e adaptativa. Outro exemplo é o dos dispositivos de borda (comumente sistemas embarcados, como AGV), que podem ter capacidades, adaptativas, de autonomia e de processamento em tempo

real em função das suas atribuições.

Operadores humanos que atuam nos processos de manufatura são indispensavelmente abrangidos pelo modelo SAMOM. Eles também devem ser representados digitalmente por seus respectivos agentes.

### **3.2.1 Classes de Agentes**

Os DIM são divididos em classes por semelhança de características e de papel no sistema de manufatura. Os dispositivos físicos, naturalmente, já possuem diversas classificações estabelecidas a partir de suas características e funções, como é o caso de máquinas ou operadores. Estas ainda podem ter outros níveis de classificações, a partir de aspectos como conversão de energia (motores) ou transformação direta de materiais por meio da movimentação de ferramentas (máquinas ferramenta). Outro exemplo são os transportadores, que são classificados por sua função de transportar, podendo ser estes aplicados em diferentes situações, de diferentes formas, por meio de dispositivos de diferentes origens (esteiras, automóveis, AGV etc.)

Com isso, é necessário então estabelecer também classificações para os agentes, as quais não necessariamente deverão ser as mesmas dos dispositivos físicos. Além disso, a necessidade de classificação se dá também pelo fato de que os agentes serão modelados e, logo, demandam padrões para tal. Contudo, as classificações dos agentes devem ser coerentes às dos dispositivos físicos devido ao fato de que cada dispositivo físico possui seu respectivo par agente, e ambos formam uma unidade (o DIM).

Uma tipificação prévia à classificação dos agentes é a de agentes reativo e cognitivo (ou inteligente). Os agentes do tipo reativo fazem apenas operações simples (como adquirir um dado, ou verificar uma variável por meio uma condição) e sem caráter de inferência ou predição. Por outro lado, os agentes do tipo cognitivo processam informações com base em modelos matemáticos, estatísticos ou de aprendizado. Esses modelos podem ser configurados ou treinados de forma assíncrona e importados para os agentes durante a criação destes. É possível também que os agentes sejam criados de forma que seus modelos sejam reconfigurados ou retreinados automática e sincronamente.

O modelo SAMOM propõe a utilização do diagrama de classes em Linguagem



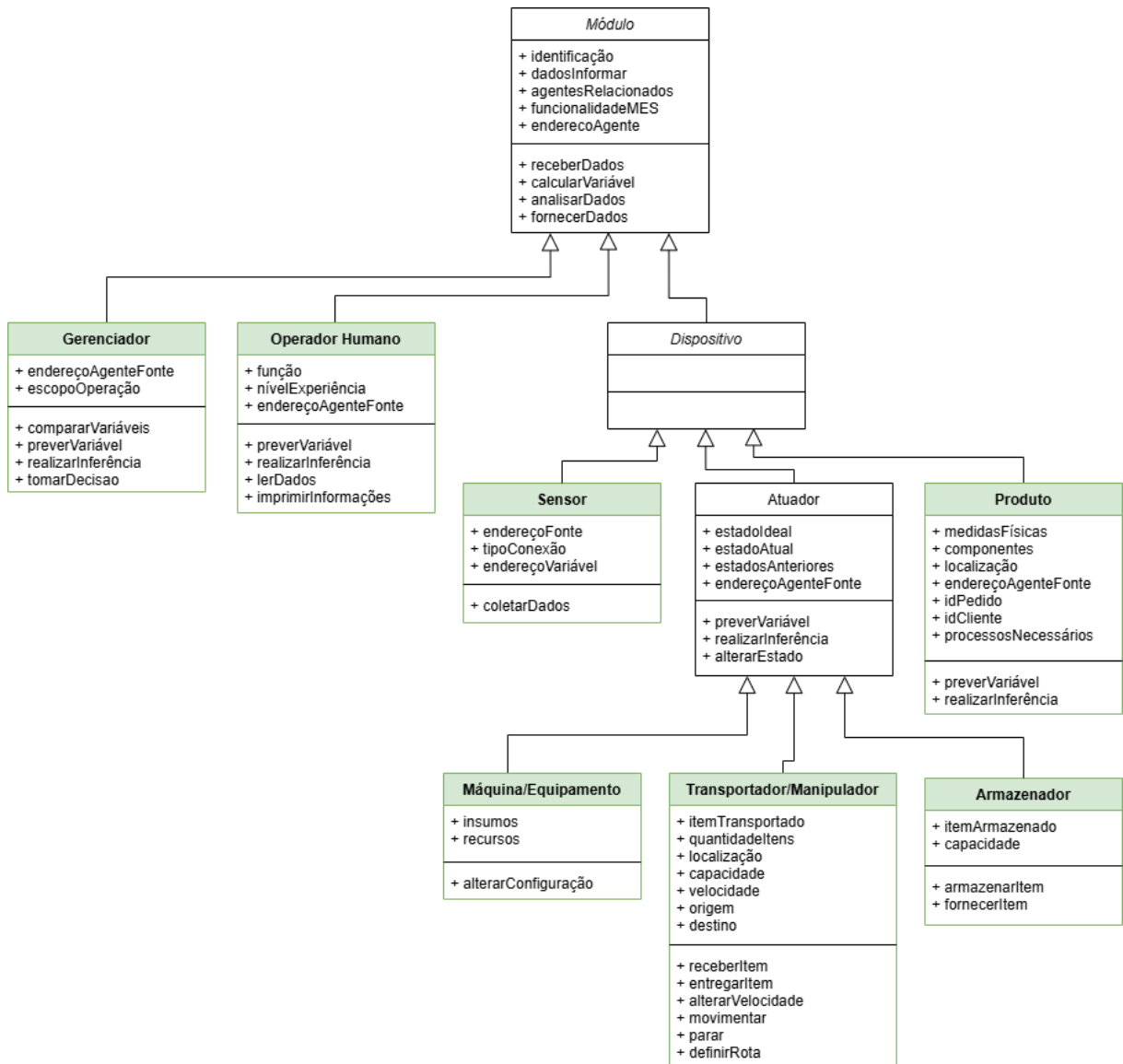
de Modelagem Unificada (UML, do inglês, *Unified Modeling Language*), um método que pode ser usado para estabelecer as classes de agentes e modelar o SMA. Na Figura 3.4 é apresentado um diagrama UML proposto como base para implementar classificações específicas de acordo com cada caso de uso. Estes casos de uso podem ter mais níveis de herança, outras classes diferentes das sugeridas, como também outros tipos de relações entre classes além da herança (por exemplo: dependência e agregação). No diagrama, cada bloco, que corresponde cada classe de agente, comporta o nome (retângulo superior), os atributos (retângulo central) e os métodos (retângulo inferior) de cada classe. Os atributos dizem respeito a propriedades que determinadas classes de agentes possuem intrinsecamente, enquanto que os métodos tratam de funções e ações.

A principal classe na Figura 3.4 é a denominada Módulo. As demais subclasses herdam seus atributos e métodos, isto é, qualquer agente que venha ser instanciado pode possuir tais atributos (identificação, dados a serem armazenados temporariamente e compartilhados, outros agentes relacionados, funcionalidades MES relacionadas e endereço do próprio agente) e métodos (receber, analisar e fornecer dados, bem como executar lógicas e cálculos).

A classe Sensor, que é uma subclasse da classe Dispositivo, tem o papel de apenas coletar dados e fornecer para outros agentes. Esta é intrinsecamente uma classe de agentes reativos. A classe Produto, que também é uma subclasse da Dispositivo, tem o papel de processar e fornecer informações relacionadas ao produto físico.

A classe Atuador representa digitalmente dispositivos físicos que atuam nos processos de manufatura. As classes Máquina/Equipamento, Transportador e Armazenador são herdeiras da classe Atuador. A classe Armazenador tem o papel de representar digitalmente dispositivos físicos de armazenamento de produtos ou materiais (insumos), tanto localmente (para apenas uma estação ou unidade produtiva) quanto globalmente (para várias estações, unidades produtivas, processos ou filiais). Semelhantemente, a classe Transportador representa digitalmente dispositivos físicos de transporte, movimentação e manipulação de produtos ou materiais.

Figura 3.4 Classes de agentes



Fonte: Próprio Autor.

A classe Máquina/Equipamento representam digitalmente diferentes tipos de dispositivos físicos que atuam nos processos de manufatura como, bombas, motores, tornos mecânicos etc. Recomenda-se que esta classe seja base para mais níveis de subclasses, se pertinente à aplicação, com o intuito de estabelecer os mais adequados papéis em comum dos agentes que representam diferentes tipos de dispositivos físicos das categorias de máquinas, ferramentas, equipamentos etc.

A classe Operador Humano representa digitalmente seres humanos que atuam no processo de manufatura. Esta tem o papel de processar informações a partir de

Interfaces Homem-Máquina (IHM) além de informações sobre as atividades destes próprios operadores.

Agentes da classe Gerenciador podem compor a representação digital de qualquer tipo de dispositivo físico (nos casos em que tal dispositivo requeira mais de um agente). Estes também podem compor a representação digital de mais de um dispositivo (nos casos em que seu escopo de operação alcance os processos de uma estação, linha, planta etc. que possuam mais de um dispositivo físico). A classe Gerenciador tem o papel de tomar decisões a partir de diferentes informações, provenientes de diferentes agentes e/ou processadas por si mesma.

### **3.3 SISTEMA DE EXECUÇÃO DA MANUFATURA**

O MES proposto pelo modelo SAMOM implementa as seguintes principais funcionalidades, com base nas apresentadas nos modelos MESA (McClellan; Weaver, 2011) e por Jaskó e colaboradores (2020), são:

- a) aquisição e gerenciamento de dados: coleta, armazenamento, consultas, fornecimento e troca de dados da manufatura;
- b) acompanhamento e rastreamento da produção: localização, identificação de estado atual, histórico e demais informações intrínsecas de produtos e recursos de manufatura;
- c) alocação e estado de recursos: planejamento ideal da produção a partir da disponibilidade de recursos;
- d) programação detalhada da produção: reunião das informações necessárias para agendamentos que garantam a usabilidade ideal de recursos e dispositivos;
- e) gerenciamento de processos: determinação e direcionamento de processos específicos, ou da quantidade ideal de processos, necessários para o produto, de acordo com determinado pedido;
- f) análise de desempenho: monitoramento do estado atual de produção e avaliação de indicadores por meio de interfaces gráficas;
- g) despacho de unidades de produção: determinação do fluxo ideal de produtos e das velocidades de operação das estações de manufatura para controle de

- ritmo da produção, de acordo com o ritmo de mercado, e redução de desperdícios;
- h) historiamento de processos: registro de variáveis de manufatura com seu respectivo instante de coleta (tempo);
  - i) análise de dados e IA: aplicação de modelos estatísticos e de ML para previsibilidade de eventos e geração de informações e conhecimento, a partir de dados de manufatura.

Para cada caso de implementação, podem ser necessárias diferentes funcionalidades, inclusive diferentes das indicadas. Para o caso de aplicações com maior abrangência, funcionalidades como gestão da qualidade ou gestão da manutenção poderiam fazer parte.

A partir das funcionalidades apresentadas, e por meio de seus conceitos e regras, o sistema MES tem a função de orquestrar o sistema de manufatura. Cada funcionalidade reúne ou processa informações e atividades de uma determinada área de G&C da manufatura. É recomendado pelo modelo SAMOM que o sistema MES possibilite que suas funcionalidades estejam disponíveis para serem solicitadas, independentes das outras funcionalidades, como serviços. Além disso, é essencial, para a caracterização do modelo SAMOM, que o MES esteja apto a receber informações dos agentes, de forma que estas sejam somadas com as informações próprias do MES e novas informações mais ricas (como as preditivas) de G&C sejam geradas. Estes quesitos configuram e ressaltam a importância do paradigma SOA para o modelo proposto.

Sendo assim, os DIM solicitam o serviço e enviam e recebem informações das funcionalidades específicas necessárias para a sua atuação no sistema de manufatura. Enquanto a modelagem específica de cada agente é essencial para estabelecer as habilidades preditivas e adaptativas sob um ponto de vista individual de cada DIM, a cooperação e coordenação do SMA, bem como a orquestração do sistema MES, são essenciais para estabelecer as habilidades adaptativas e preditivas – além das características de ser distribuído, descentralizado e heterárquico – ao sistema de manufatura como um todo.

Dentre as funcionalidades apresentadas, destacam-se como essenciais as funcionalidades de Análise de dados e IA e Historiamento de processos:

- a) análise de dados e IA contribui com a atribuição de habilidades preditivas e adaptativas ao sistema de manufatura;
- b) historiamento de processos possibilita a existência do tipo de dados (historiados) mais pertinentes para realização de inferências, previsões e adaptações, no contexto da G&C da manufatura.

Outros sistemas podem cooperar e complementar o MES proposto no modelo SAMOM, de acordo com as funcionalidades que possuem. Como exemplos, os sistemas SCADA (que tem como alguns de seus papéis coletar, mostrar dados de processos e controlá-los) e de gerenciamento de banco de dados (que armazenam e permitem consulta de dados), comporiam funcionalidades relacionadas a aquisição e gerenciamento de dados.

O MES, bem como estes outros sistemas que podem o complementar, se aplicável, possuem também o papel de ser a interface de interação do SAMOM para o usuário, principalmente gestores e operadores. A implementação de outras formas de IHM para o modelo SAMOM podem se apresentar mais adequadas. Entretanto, tal investigação não faz parte do escopo do presente trabalho.

### **3.3.1 Contexto de Gestão e Controle e Relações de Tempo**

Dois dos principais guias para a implementação do modelo SAMOM num sistema de manufatura são os indicadores utilizados – também conhecidos como indicadores chave de desempenho (KPI, do inglês *Key Performance Indicators*) – e as decisões tomadas. Estes dois são predeterminados principalmente a partir de estratégias de negócio, políticas corporativas, cultura organizacional e metodologias de gestão. Sendo assim, a escolha destes dois quesitos estabelece um contexto para a G&C do sistema. As decisões são tomadas a partir do acompanhamento dos KPI. Como exemplo, estes são alguns KPI comuns em MES:

- a) tempo de ciclo – tempo de duração de um processo único realizado sobre um produto;
- b) taxa de produção (ou taxa de saída) – quantidade de produtos produzidos por período;
- c) taxa de produtos retrabalhados (ou descartados) – quantidade de produtos retrabalhados (ou descartados) por período;

- d) *takt time* – tempo demandado pelo cliente para a produção de um produto.
- e) custo de produção – custo monetário ou quantitativo de insumos necessários para a produção de um produto ou em um determinado período;
- f) tempo médio entre falhas – tempo em que uma máquina opera entre uma falha e outra seguinte, em média;
- g) eficiência global do equipamento (OEE, do inglês *Overall Equipment Effectiveness*) – multiplicação entre os indicadores disponibilidade, desempenho e qualidade;
- h) disponibilidade – tempo em que uma máquina está disponível para operação, em relação ao tempo total;
- i) desempenho – quantidade de produtos processados por uma máquina, em relação à capacidade teórica de produção desta máquina, por período;
- j) qualidade – quantidade de produtos processados com sucesso por uma máquina, em relação à quantidade total processada (incluindo os produtos defeituosos processados);
- k) estado atual da máquina (ou estação) – se disponível, em operação (ocupado) ou indisponível (em manutenção ou com falha).

Com este contexto de G&C estabelecido, é possível determinar quais indicadores podem ser calculados, inferidos ou preditos e quais decisões podem ser tomadas. Direcionados por estes indicadores e decisões o SMA é modelado (determinação dos agentes, seus papéis e as devidas comunicações), os agentes são construídos (programação de lógicas, seleção de dados, criação e treinamento de modelos, endereçamentos de comunicações). Assim são estabelecidas as ações preditivas e adaptativas dos DIM e, conseqüentemente, o modelo SAMOM.

A determinação dos indicadores e das decisões, bem como o estabelecimento dos agentes, impacta na determinação das relações de tempo da operação do SAMOM. Estas relações referem-se à determinação de sincronicidade da operação dos agentes. Em outras palavras, referem-se à determinação de em quanto tempo um agente deve responder ou decidir (se o mais rápido possível – tempo real/sincronamente – ou com determinado atraso). Referem-se também ao período considerado no cálculo dos indicadores e para as tomadas de decisão.

Diante então do contexto de G&C predeterminado em uma empresa, escolhe-se

qual período deve ser utilizado nos indicadores e, conseqüentemente, como cada agente deve operar em relação ao tempo. O exemplo a seguir ilustra esta situação. Uma empresa de linha de montagem de bicicletas monitora os indicadores taxa de produção, custo de produção, qualidade e taxa de produtos descartados, utilizando um turno como período. Um agente da classe Atuador determina a predição da quantidade de arame (custo de produção), apenas uma vez por turno, necessária para a montagem dos raios nas rodas, bem como da qualidade desta estação. Outro agente, da classe Gerenciador monitora instantaneamente a quantidade de arame disponível em estoque e, com a informação preditiva do agente anterior, pode decidir entre realizar ou não um pedido de reposição de arame ao fornecedor antes do estoque esgotar. Um terceiro agente, também da classe Gerenciador, monitora a taxa de rodas descartadas, recebe a predição da qualidade do primeiro agente, a decisão do segundo agente e pode decidir se inicia um processo de retrabalho para uma das rodas descartadas (visto que estas podem não precisar de novo arame) enquanto novas rodas a serem montadas poderiam não ser finalizadas (por falta de arame).

Toda essa dinamicidade do exemplo acima, requer a operação de cada agente em ciclos de diferentes intervalos. O primeiro agente opera uma vez por turno. O segundo está em operação síncrona, mas, paralelamente, sua ação pode acontecer em um tempo maior ou nem mesmo acontecer. O terceiro agente também está em operação síncrona e sua decisão é dependente da decisão do segundo. As relações de tempo também implicam em prioridades e dependências de decisão, que por sua vez estão relacionadas com regras de comunicação, discutidas na seção seguinte.

### **3.4 COMUNICAÇÃO**

A comunicação M2M é um paradigma essencial ao modelo SAMOM, pois torna possível a comunicação direta entre qualquer componente do sistema de manufatura. Contudo, para cada caso de instanciação, a depender do seu estágio de maturidade, é necessário que sejam determinadas regras para estabelecer um funcionamento efetivo do modelo e evitar conflitos na comunicação e integração entre sistemas. Cada caso deve possuir regras próprias, entretanto, abaixo está descrito alguns princípios para direcionar tais regras.

Para os casos de sistemas mais maduros e componentes originalmente capazes de serem descentralizados, distribuídos e heterárquicos, o SMA é condicionado para ter maiores capacidades de cooperação e coordenação. Dessa forma, os agentes são autônomos o suficiente para tomar decisões e evitar multiplicidades de comandos enviados aos dispositivos físicos. Em outras palavras, os agentes são suficientemente inteligentes para perceber qual deles (ou mesmo o MES) possui prioridade para enviar comandos aos dispositivos físicos, em determinados momento e contexto. O paradigma SMA possibilita que, com agentes, os próprios

Para os casos de aplicação nos quais o estágio de maturidade é menos elevado, com componentes e sistemas legados, sob arquiteturas centralizadas e hierarquizadas, o modelo SAMOM é capaz de atribuir habilidades adaptativas, porém a comunicação deve ser configurada de forma a permitir uma operação adequada. Por exemplo, a comunicação entre um CLP e uma máquina geralmente é do tipo mestre-escravo. Neste caso, os agentes não conseguem comunicar diretamente com a máquina em paralelo ao CLP, pois existiria duplicidade de controle. Logo, é necessário que o agente se comunique com o CLP (possível de acontecer pelo fato da comunicação entre um sistema e um CLP geralmente ser do tipo cliente-servidor), o que estabelece determinadas centralização e hierarquia no controle. Mesmo com tais limitações, o agente oferece um controle inteligente, baseado em informações preditivas e/ou de diversos outros agentes e do MES.

A depender do contexto, principalmente em sistemas menos maduros, alguns comandos podem permanecer com o controle tradicional manual. Em paralelo, neste mesmo sistema, algumas outras variáveis são controladas pelos agentes que vão comandar diretamente os dispositivos físicos (ou o próprio MES, de forma que se segue a pirâmide de controle até alcançar os dispositivos físicos). O presente modelo sugere que, o tanto quanto for possível, os dispositivos físicos sejam sujeitos a comando apenas do SMA, enquanto o MES teria somente o papel de leitura de dados e disponibilização de informações. Em outras palavras, o SMA age como o controlador e o MES como processador de informações e visualizador. A solicitação, pelos agentes, das funcionalidades MES, permite que estes atuem sobre seus respectivos dispositivos físicos de acordo com orquestração do sistema MES.

As Figuras 3.2 e 3.3 demonstram a possibilidade de implementar o modelo



SAMOM com base nas alternativas descritas acima. Entre um agente genérico e funcionalidades MES, a comunicação é mandatória tanto para escrita quanto para leitura. Sem ao menos um dos agentes ter acesso a informações contextualizadas pelo MES (mesmo que assíncronas) e enviar informações ao próprio MES, a aplicação não é caracterizada como sendo do modelo SAMOM. Por outro lado, entre um dispositivo físico genérico e funcionalidades MES, a comunicação é mandatória apenas num sentido (obtenção de sinais e dados pelo MES). O sentido inverso deve ser implementado apenas se o caso de uso necessitar.

As alternativas de protocolos de comunicação a serem aplicados também são fundamentais no modelo SAMOM, pelo fato de poder limitar sua implementação. É recomendado o padrão OPC UA para as comunicações da arquitetura do SAMOM, principalmente as que envolvem os dispositivos físicos, pois possui maior capacidade de interoperabilidade em relação a demais protocolos.

É possível também a implementação de outros protocolos modernos, como MTConnect ou MQTT, como também outros mais legados, como Modbus, OPC Classic ou EtherNet/IP. Contudo, a depender do caso de aplicação, estes protocolos podem limitar a interoperabilidade entre os dispositivos e, conseqüentemente requerer elementos centralizadores, como *Gateways* ou API. Este fato não necessariamente impede o modelo SAMOM de operar, mas o limita quanto aos seus objetivos e, conseqüentemente, de possuírem maiores habilidades adaptativas.

Para alguns casos, a depender da natureza do dispositivo ou do MES, são aplicáveis outros protocolos que operam sob a solução *Web Services*<sup>6</sup> como, por exemplo, o Protocolo de Transferência de Hipertexto (HTTP, do inglês *Hypertext Transfer Protocol*) e o WebSocket, sem comprometer os objetivos em questão, por esta ser orientada a serviços. Estes protocolos *Web* requerem a definição do seu estilo arquitetural (conjunto de princípios e restrições para sua construção e/ou operação), que podem ser REST ou o Protocolo de Acesso a Objetos Simples (SOAP, do inglês *Simple Object Access Protocol*), por exemplo. Requerem a definição também da sua linguagem de transporte, que podem ser a Linguagem de Marcação Extensível (XML,

---

<sup>6</sup> Comunicações que utilizam a internet, ou uma rede específica de computadores, e suas ferramentas tradicionais, como navegadores e endereços URLs, sob pilhas também tradicionais de protocolos (como o TCP/IP e o modelo OSI).

do inglês *Extensible Markup Language*), a Notação de Objeto JavaScript (JSON, do inglês *JavaScript Object Notation*) ou a Linguagem de Comunicação de Agente da Fundação para Agentes Físicos Inteligentes (FIPA-ACL, do inglês *Foundation for Intelligent Physical Agents – Agent Communication Language*), por exemplo. Para as comunicações entre agentes e agentes com o MES, a aplicabilidade de tais protocolos *Web* é maior, devido à natureza dos componentes envolvidos (computadores convencionais), bem como a orientação a serviços desta comunicação e o quanto esta contribui para alcançar os objetivos de coordenação e cooperação entre os agentes.

*Este capítulo apresenta a instância de validação do método proposto e os detalhes importantes envolvidos na aplicação.*

## **VALIDAÇÃO DO MODELO**

Nesta seção é apresentada a validação do modelo SAMOM em uma instância de sistema de manufatura. O local onde foi realizada tal instanciação é apresentado, além de configurações, componentes e tecnologias utilizados. São detalhados também como foram dispostos os sistemas DIM, SMA e MES para a G&C deste sistema de manufatura. Esta validação visa demonstrar a capacidade de um sistema de manufatura, sob a arquitetura do modelo SAMOM, ser distribuído, descentralizado, heterárquico e orientado a serviços. Visa também demonstrar as habilidades preditivas e adaptativas deste sistema, apresentando resultados que possam evidenciar tais demonstrações.

### **4.1 A PLANTA DE MANUFATURA AVANÇADA**

O sistema de manufatura sobre o qual será implementado o modelo SAMOM fabrica e opera alguns processos de tratamento finais de tampas e bases para cilindros pneumáticos. Esta planta tem como objetivo ser um piloto para sistemas de manufatura 4.0 e possui caráter didático e demonstrativo (Figura 4.1). Ela é denominada de Planta de Manufatura Avançada (PMA) e está localizada na cidade

de Salvador, na instituição SENAI Cimatec<sup>7</sup>. A PMA é composta por seis estações: Usinagem, Entrada, Transporte, Inspeção/Furação, Limpeza e Saída.

Figura 4.1 Planta de Manufatura Avançada (PMA) do SENAI Cimatec

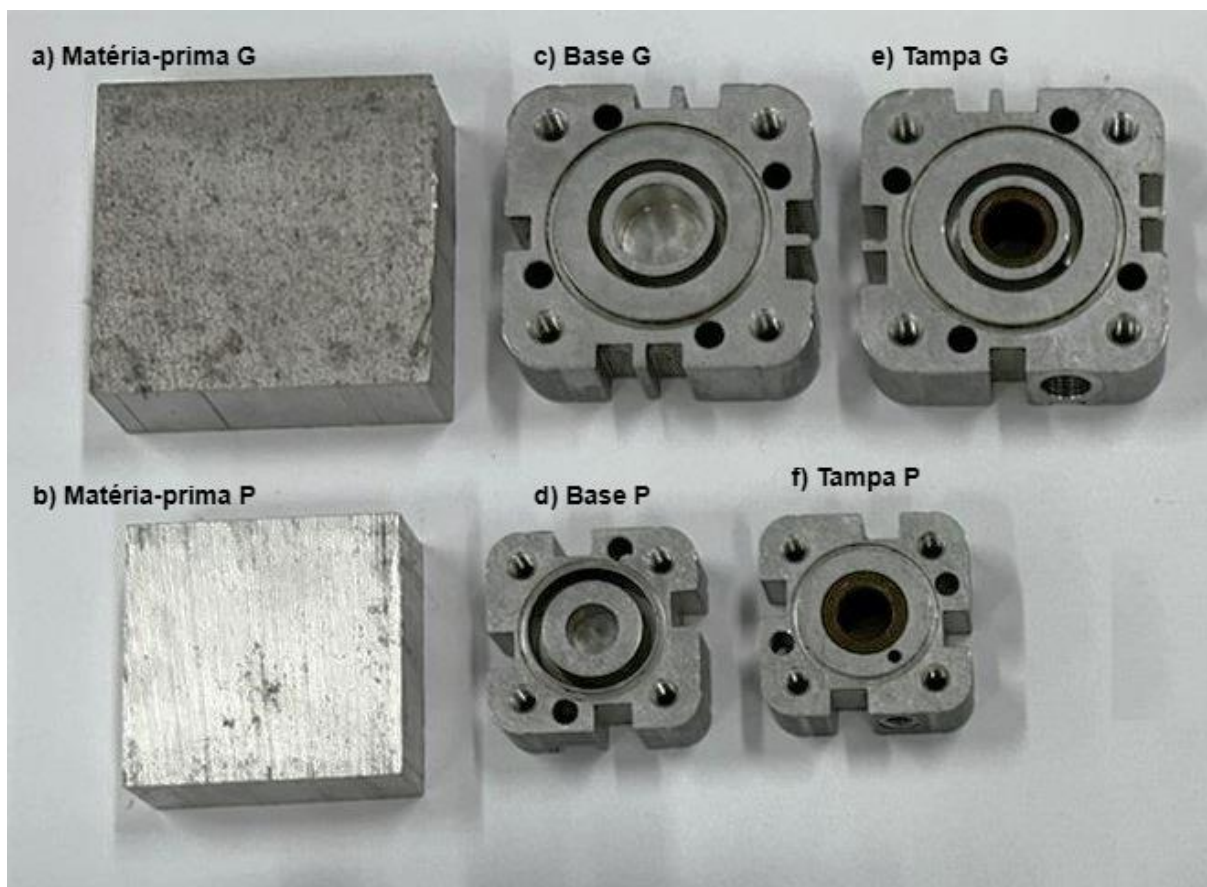


Fonte: Próprio Autor.

<sup>7</sup>A execução da instância do modelo SAMOM realizada na PMA, bem como sua construção (sistemas DIM, SMA e MES), tiveram a participação de equipes que atuam na PMA: técnicos, especialistas, professores e pesquisadores de iniciação científica, pós-graduação e de projetos internos da instituição.

As tampas e bases (peças) são de formato quadrado e possuem duas variações de tamanho quanto a dimensão de lado: P (25mm) ou G (40mm) (Figura 4.2). O cliente da PMA, a Fábrica Modelo, que realiza a montagem de cilindros pneumáticos, solicita pedidos de tampas e bases à PMA. As tampas, diferentemente das bases, possuem uma bucha de latão montadas em si por ajuste e um anel polimérico de vedação.

Figura 4.2 Peças fabricadas na PMA



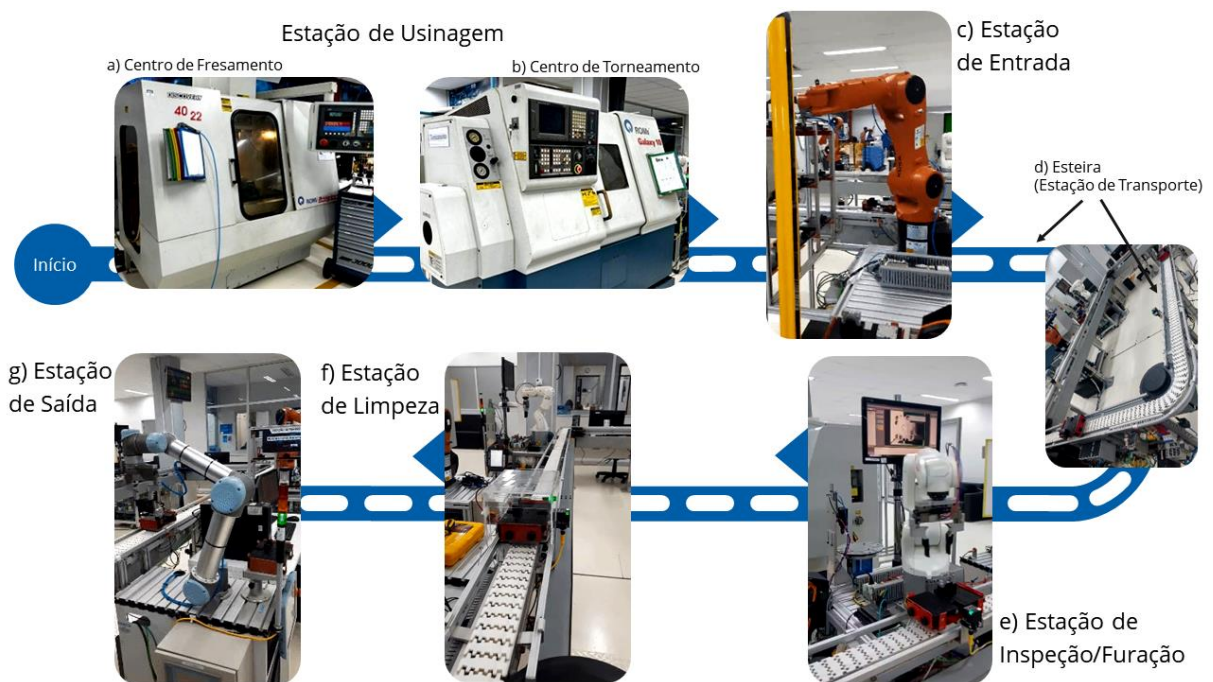
Fonte: Próprio Autor.

O processo na PMA (Figura 4.3) é iniciado na estação de Usinagem. Um operador coloca a matéria prima - paralelepípedos de alumínio (Figura 4.2 a) e b)) – num Centro de Fresamento (Figura 4.3 a)), que atua por Controle Numérico Computadorizado (CNC, do inglês *Computer Numerical Control*), e aciona o devido programa para realizar os processos de fresamento na peça. Em seguida o operador leva a peça até o Centro de Torneamento CNC (Figura 4.3 b) e aciona o devido programa para realizar os processos de torneamento. A estação de Usinagem



atualmente não possui conexões diretas e em tempo real de sinais, dados e informações com as demais estações. Tais conexões estão sendo analisadas para futura integração das máquinas CNC. O principal desafio trata do fato de que os sistemas de tais máquinas são legados e fechados à comunicação em tempo real por protocolos industriais. Esta situação não faz parte dos objetivos do presente trabalho, mas o afeta. Devido à não integração da estação de Usinagem, esta não é abrangida pelo escopo da instanciação do modelo SAMOM na PMA.

Figura 4.3 Fluxo do processo da PMA

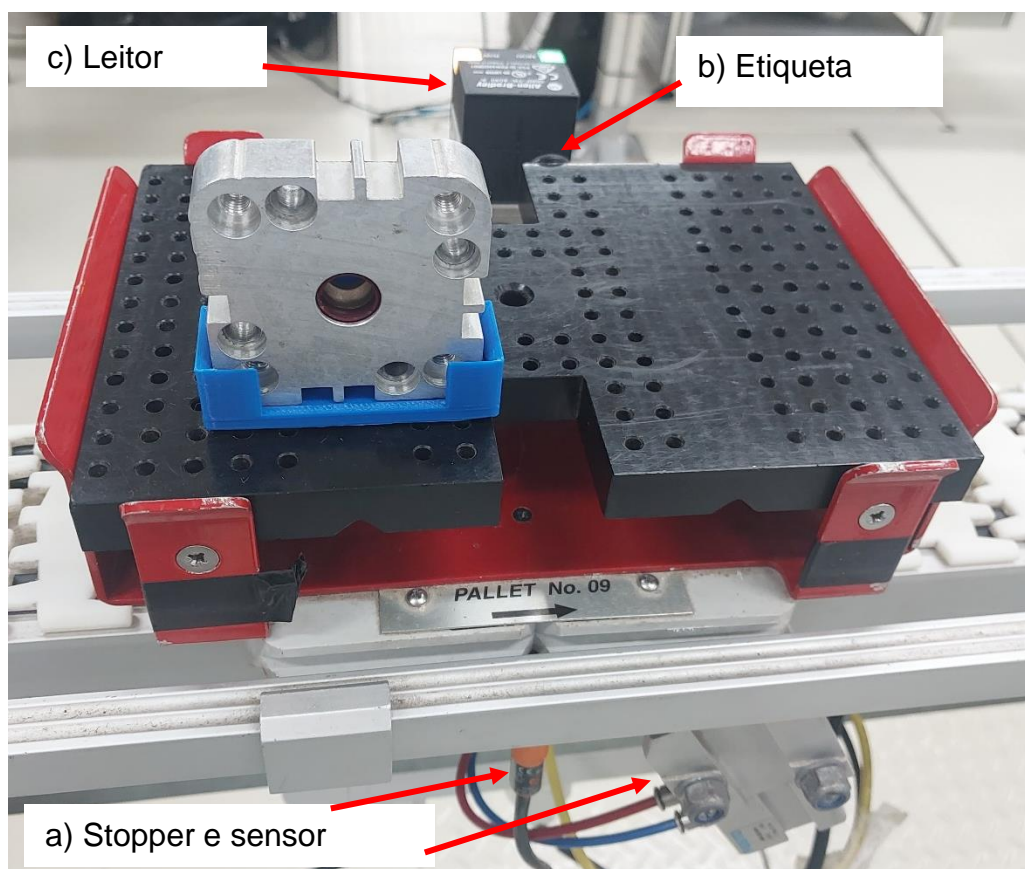


Fonte: Próprio Autor.

Após a Usinagem completa, o operador leva a peça até a próxima estação a de Entrada (Figura 4.3 c)). Ele posiciona a peça num compartimento, um robô a apanha e a armazena em uma estante, que atua como *buffer* do processo. Este mesmo robô transfere uma peça por vez do *buffer* para um palete (Figura 4.4), que se desloca sobre uma esteira transportadora da próxima estação, a de Transporte (Figura 4.3 d)). Esta esteira é retangular, de caminho fechado (cíclico), e transporta os paletes com as peças entre as demais estações. Os paletes passam pela estação de Entrada, em seguida pela estação de Inspeção/Furação, depois pela estação de Limpeza, também pela estação de Saída e retorna para a estação de Entrada.

Cada uma dessas estações possuem um bloqueador pneumático (*stopper*) (Figura 4.4 a)), que pode interromper um palete, reconhecido por um sensor indutivo, caso a peça necessite passar por aquela estação. Uma dessas estações identifica se uma peça passará por seu processo por meio do RFID. Cada palete possui uma etiqueta de RFID (Figura 4.4 b)) que carrega informações da peça que foi depositada nele pelo robô da estação de Transferência. Cada estação possui um leitor de RFID (Figura 4.4 c)).

Figura 4.4 Pallet com peça na esteira



Fonte: Próprio Autor.

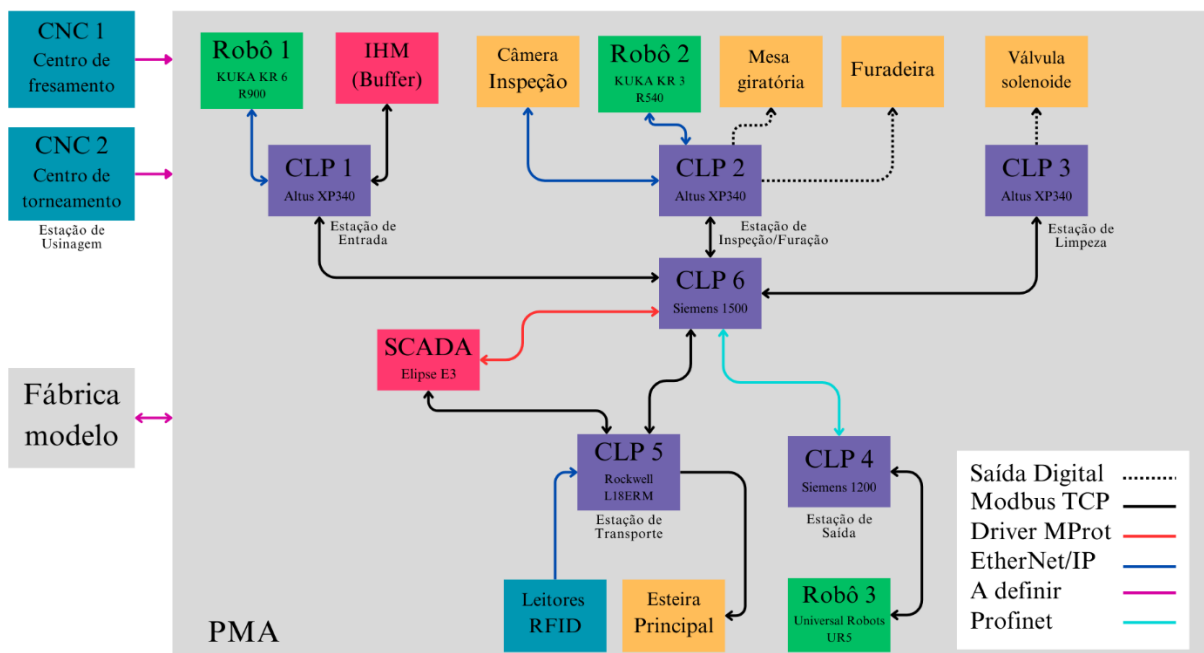
Na estação de Inspeção/Furação uma peça que, de acordo com a personalização do pedido, necessite de inspeção ou de furação, ou de ambas, é apanhada do paleta por um outro robô (Figura 4.3 e)). Este põe a peça em uma mesa giratória que a posiciona para ser fotografada por uma câmera e inspecionada por um sistema de visão computacional – que identifica e registra possíveis inconformidades

na peça. A mesa giratória também posiciona a peça para ser submetida a uma simulação de furação por meio de uma furadeira. Após um dos processos, ou ambos, o mesmo robô apanha a peça na mesa e a põe em um palete.

A estação de Limpeza consiste numa câmara acrílica (Figura 4.3 f)), posicionada na própria esteira, e um bocal de saída de jato de ar. O palete que leva a peça que passará pelo processo de limpeza (longa ou curta), é parado pelo *stopper*. Então uma válvula direcional de acionamento por solenoide recebe o comando para que o jato de ar seja liberado. Na estação de saída, um terceiro robô (Figura 4.3 g)) apanha uma peça em um palete que foi parado e a coloca em um compartimento. Um operador recolhe a peça deste e a libera para ser entregue à cliente da PMA (Fábrica Modelo).

O processo descrito acima é operado por meio da arquitetura da Figura 4.5, aqui denominada e Arquitetura Convencional. A arquitetura traz os controladores, sistemas, protocolos de comunicação e principais atuadores.

Figura 4.5 Arquitetura Convencional da PMA



Fonte: Próprio Autor.



## 4.2 CONTEXTO REPRESENTATIVO E CRITÉRIOS DE VALIDAÇÃO

Para que o modelo SAMOM seja instanciado sobre o sistema descrito na seção 4.1 e a validação seja realizada, foi necessário estabelecer um contexto representativo de G&C. Este trata da determinação dos indicadores e decisões de G&C a serem respectivamente monitorados e tomadas. Tal determinação se fez necessária para delimitar um escopo suficiente para representar a validação desejada.

Os indicadores escolhidos foram a disponibilidade das estações, a taxa de saída e o *takt time*. O primeiro é um indicador que geralmente compõe o OEE, é definido pela Equação 1 e representa o quanto que uma máquina esteve disponível em um determinado período. O segundo representa a quantidade de produtos finalizados por um processo em um determinado período e é definido pela Equação 2. O terceiro representa o ritmo necessário para que o processo atenda a demanda de mercado e é definido pela Equação 3.

$$d = \left( \frac{t_t - t_p}{t_t} \right) / p \quad (1)$$

$$ts = \frac{q_p}{p} \quad (2)$$

$$tk = \left( \frac{t_{ol}}{q_d} \right) / p \quad (3)$$

Sendo:

- $d$  a disponibilidade de uma estação, medida por um valor adimensional entre 0 e 1;
- $t_t$  o tempo total em que a estação esteve em operação;
- $t_p$  o tempo total em que a estação esteve indisponível devido a paradas de manutenção e/ou falha;
- $p$  o período considerado (definido para todos os casos do presente trabalho como 1 dia);
- $ts$  a taxa de saída, medida em unidades produzidas por período (*und/dia*);
- $q_p$  a quantidade produzida;
- $tk$  o *takt time*, medido em segundos por unidade (*s/und*);
- $t_{ol}$  o tempo total em que o processo esteve em operação (tempo operacional líquido);

- $q_d$  a quantidade demandada pelos clientes.

Uma vez que o contexto representativo foi definido, foram determinados critérios de comparação. Estes possibilitam a apresentação de resultados que comprovam o alcance dos objetivos da validação. Tais resultados serão demonstrados por meio de comparação entre a operação do sistema de manufatura sob sua arquitetura convencional e sob o modelo SAMOM.

O primeiro critério trata da capacidade da arquitetura ser, conjuntamente, distribuída, descentralizada e heterárquica. Este é verificado por meio das próprias disposição e configuração das arquiteturas como também a operação da PMA sob estas. O segundo critério trata do modelo ou arquitetura possuir habilidades preditivas. Este é verificado por meio do monitoramento, visualização e interação com todas as informações disponíveis relacionadas à operação da PMA em tempo real. O terceiro e último critério trata do modelo ou arquitetura possuir habilidades adaptativas. Este é verificado a partir das decisões acerca do ritmo (velocidade) de produção.

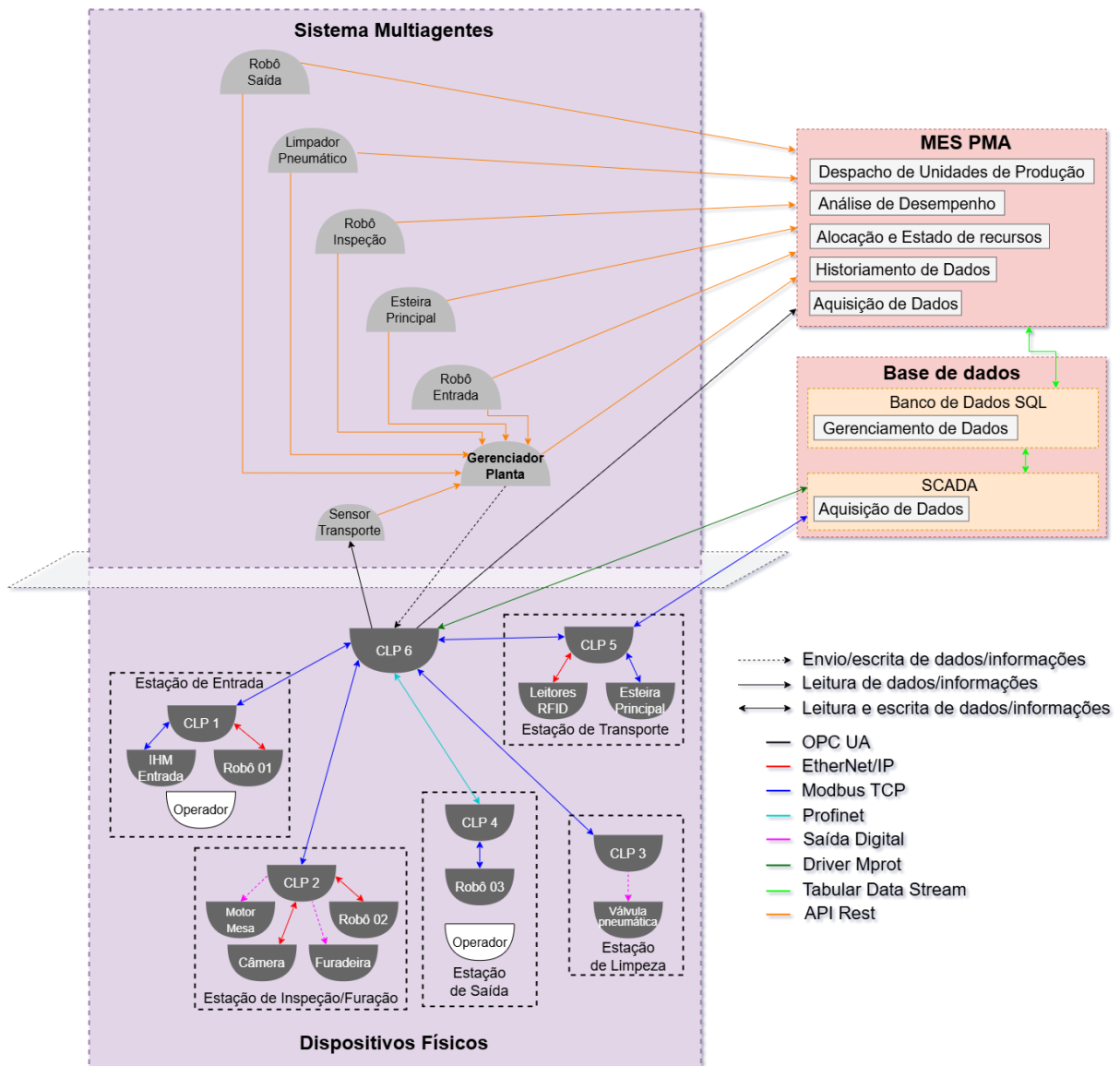
### **4.3 INSTÂNCIA DO MODELO SAMOM NA PLANTA DE MANUFATURA AVANÇADA**

A Figura 4.6 apresenta a arquitetura geral da instância do modelo SAMOM na PMA. Os DIM são representados nesta figura pelas camadas física (dispositivos físicos) e digital (SMA) para uma melhor visualização das comunicações. O MES do modelo SAMOM está representado nesta figura pela composição de um MES convencional com uma base de dados, que envolve os sistemas SCADA, e um banco de dados de Linguagem de Consulta Estruturada (SQL, do inglês *Standard Query Language*).

Na Figura 4.6 estão representadas todas as comunicações individuais entre os dispositivos físicos. Também estão representadas as comunicações destes com o MES e com os agentes (por meio do CLP 6). A Figura 4.7 traz uma alternativa de visualização da arquitetura ao apresentar os DIM da instância. Nesta figura, as representações digitais (agentes) dos dispositivos físicos estão agrupadas por estação. Existe também um agente (Gerenciador Planta) que não pertence a estações, mas faz parte da representação digital da PMA como um todo. Os CLP de

cada estação (1 a 5) controlam os dispositivos físicos e se comunicam com o um CLP central, como mostram as Figura 4.6 e 4.7. Estas também indicam quais protocolos são utilizados, bem como a natureza das comunicações, a partir das cores, do traçado e do sentido das setas.

Figura 4.6 Arquitetura Principal da instância na PMA



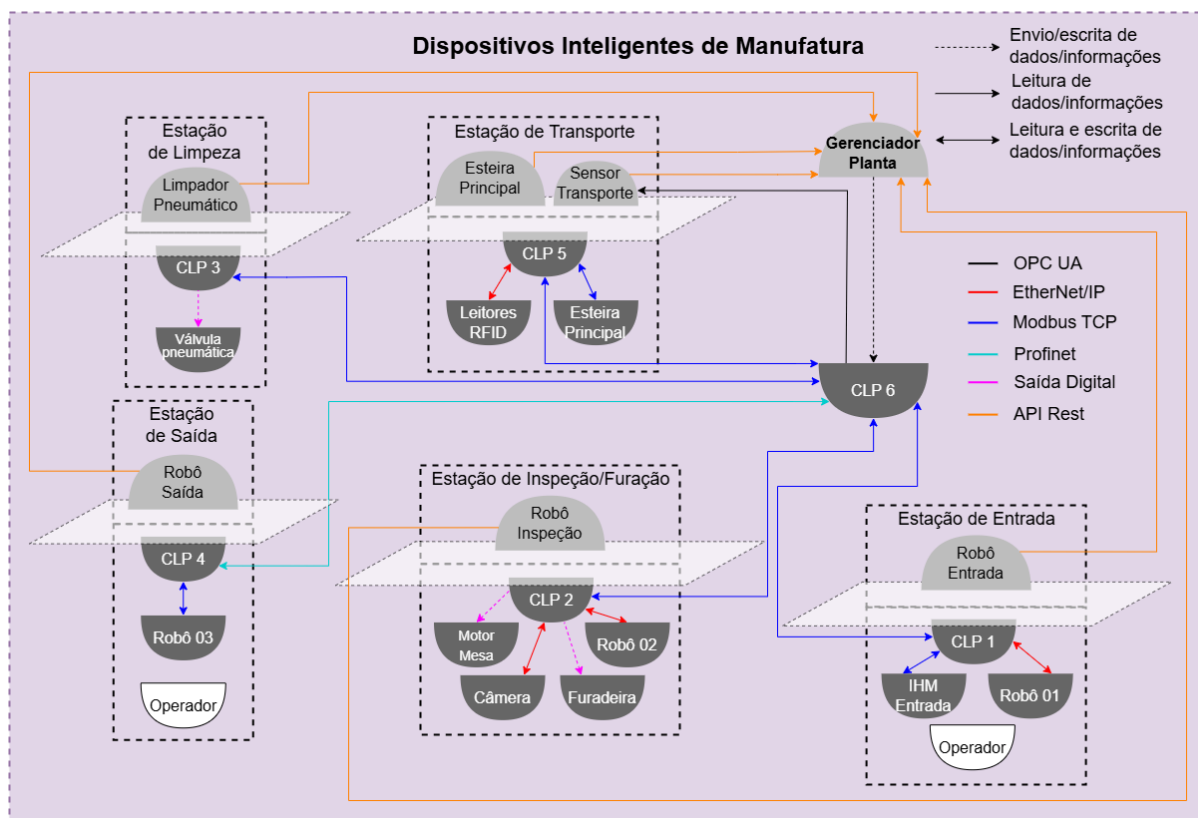
Fonte: Próprio Autor.

### 4.3.1 Sistema Multiagentes da Planta de Manufatura Avançada

A Figura 4.8 apresenta um diagrama UML análogo ao da Figura 3.4. Entretanto, ao invés de um diagrama de classes, a Figura 4.8 apresenta um diagrama de agentes

(instanciados). Nesta figura são apresentados os agentes, seus atributos e seus métodos. São apresentadas também as relações de ligação (*link*) entre os agentes.

Figura 4.7 Arquitetura dos DIM da instância na PMA

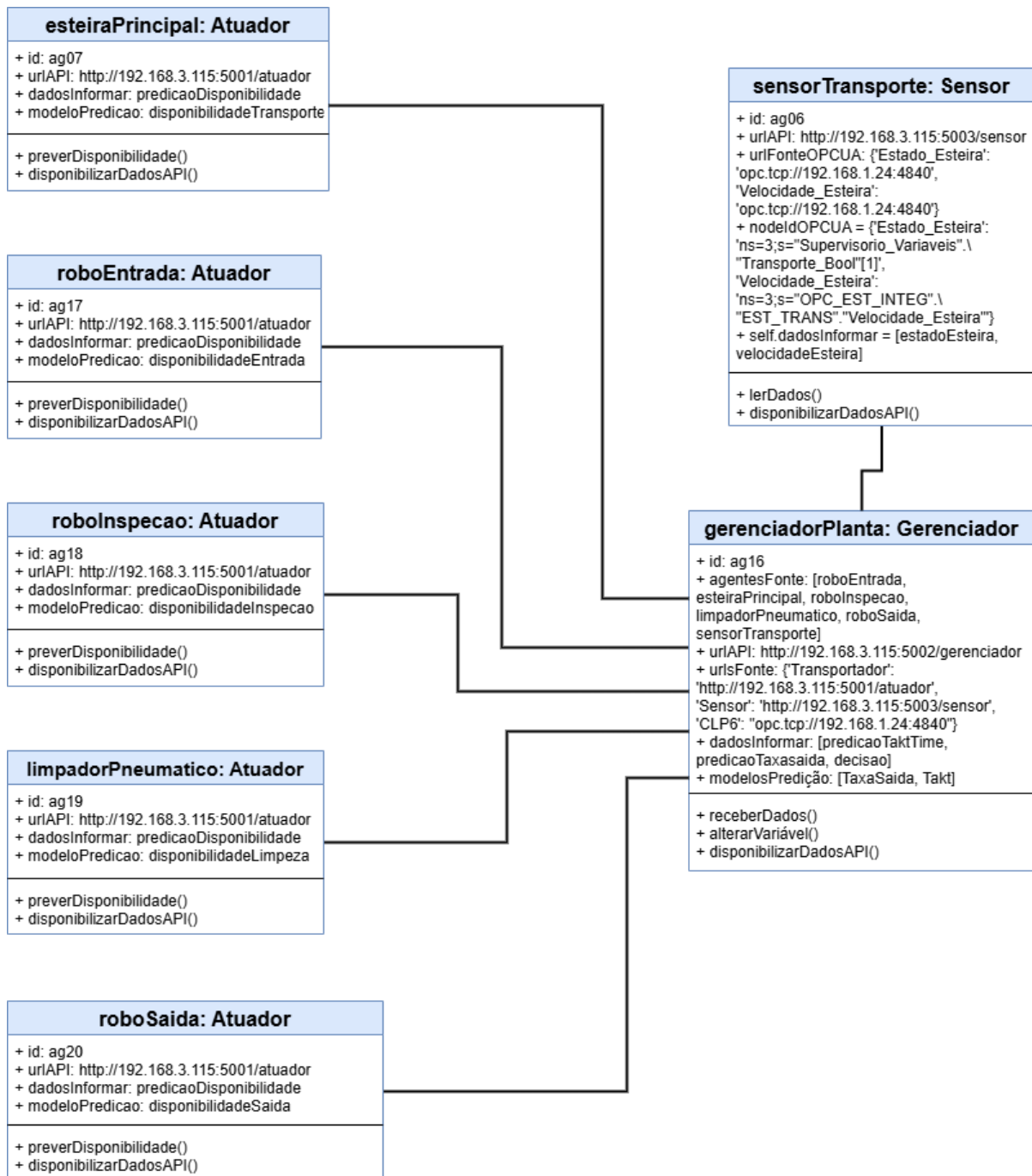


Fonte: Próprio Autor.

A estação de transporte possui um agente da classe Sensor, denominado Sensor Transporte. Este tem o papel de ler do CLP6 o estado e velocidade da esteira principal e disponibilizá-los para os demais agentes.

Os agentes da classe Atuador (Esteira Principal, Robô Entrada, Robô Inspeção, Limpador Pneumático e Robô Saída) realizam previsões das disponibilidades de cada uma das estações. Estes utilizam modelos de regressão linear que foram treinados a partir dos dados históricos. Estas previsões são disponibilizadas para o agente Gerenciador Planta, com o intuito de contribuir para que este possa tomar decisões de realizar adaptações. As previsões também são disponibilizadas para o MES, o qual as imprime em interface para que seja vista por operadores e gestores.

Figura 4.8 Diagrama UML de agentes instanciados da PMA



Fonte: Próprio Autor.

O agente da classe Gerenciador (Gerenciador Planta), com as informações preditivas que recebe, realiza a predição do *takt time* e da taxa de saída. Tal agente também utiliza modelos de regressão linear treinados por dados historiados. Também no seu algoritmo, ele calcula a previsão do ritmo de produção que a PMA manterá e

a compara com a predição do *takt time*. Se este ritmo for insuficiente para atender o *takt time*, o agente toma a decisão de aumentar a velocidade da esteira de um nível menor (40hz, proporcionais à frequência de rotação do motor elétrico que movimenta a esteira) para um nível maior (55hz). Todas as suas informações são disponibilizadas ao MES também para visualização por operadores e gestores.

Para a comunicação entre agentes foi utilizada a estrutura API REST. De acordo com Doglio (2018), REST é um estilo arquitetônico definido para a criação e organização de sistemas distribuídos, a partir de algumas restrições (cliente servidor como tipo de comunicação, interface uniforme, recursos, representações etc.) que possibilitam o alcance de algumas qualidades pelo sistema (melhor desempenho, escalabilidade, simplicidade de interface, modificabilidade, confiabilidade etc.). No caso em questão, cada agente possui uma pequena API, capaz de disponibilizar dados para diferentes tipos de aplicações (principalmente outros agentes e o MES) de diferentes linguagens. Como a estrutura foi construída em REST, foi utilizado o protocolo HTTP, que é o mais difundido neste estilo arquitetural. Como linguagem de transporte foi utilizado o JSON, também amplamente difundido. Tal estrutura foi escolhida com o intuito de serem aproveitadas as características de uma solução *Web Services*, principalmente a orientação a serviços, e de um sistema distribuído adequadamente construído. Foi também relevante na escolha sua a ampla utilização na tecnologia da informação, o que promove uma suficiente disponibilidade de informações sobre seu uso e configuração.

O SMA em questão, bem como as configurações relacionadas à sua operação, foi modelado por meio da linguagem Python. O Apêndice A apresenta os principais códigos de tais modelagem e operação. O Apêndice B contém o endereço do repositório com todos os arquivos necessários e utilizados. Os códigos das classes de agentes utilizadas estão nos arquivos “Sensor.py”, “Atuador.py” e “Gerenciador.py”. Cada classe possui também uma configuração das rotas de comunicação de seus agentes. Estes códigos estão nos arquivos “APISensor.py”, “APIAtuador.py” e “APIGerenciador.py”.

Os agentes foram instanciados em dois arquivos que, ao serem executados, colocam o sistema em operação. Um destes arquivos, o “ambiente1.py”, está relacionado com o segundo critério de validação. Neste estão instanciados todos os

agentes da classe Atuador. No outro arquivo, o “ambiente2.py”, operam os agentes Sensor Transporte, que recebe o estado e a velocidade da esteira do CLP6, e Gerenciador Planta, que recebe informações de todos os agentes, bem como envia comando de alteração da velocidade da esteira para o CLP6 e para o MES.

#### **4.3.2 Sistema de Execução da Manufatura da Planta de Manufatura Avançada**

Um dos MES utilizado na PMA é o TrakSYS da Parsec Automation, que é convencional, mas que pode ser considerado como orientado a serviços. Para a sua configuração, por exemplo, este possui serviços de lógica, gestão de dados e fluxo de trabalho, que são usados na construção personalizada de suas funcionalidades. No que se refere a sua a sua operação ele é capaz de receber dados de outros sistemas, processá-los por meio de suas funcionalidades e transformar em informações de G&C. Ele também é capaz de disponibilizar essas informações para outro Sistemas. Entretanto, para tal disponibilização é necessário acessar alguns de seus códigos para a criação de uma API. Esta é considerada como um limitador para implementação com o modelo SAMOM por ser uma centralizadora nas comunicações. Além disso, tal necessidade também se apresentou como uma dificuldade para o presente trabalho. Por outro lado, as recepções de dados e informações dos dispositivos e dos agentes, via OPC UA, bem como a configuração de algumas das suas funções, apresentaram simples configuração. Entretanto, devido ao término do contrato e ao fim da licença de uso, não foi possível operar o TrakSYS junto ao SMA.

Foi utilizado também um outro MES, o qual é objeto de um trabalho de pesquisa realizado em conjunto com o presente trabalho. Ele está em desenvolvimento e foi denominado de Sistema de Execução da Manufatura Avançada (AMES, do inglês *Advanced Manufacturing Execution System*). Seu desenvolvimento considera principalmente requisitos inerentes para a operação junto ao modelo SAMOM, em especial a orientação a serviços, e a comunicação M2M.

Para a realização da validação, o AMES foi utilizado para monitoramento em tempo real do estado das estações, monitoramento preditivo da disponibilidade das estações, monitoramento do ritmo de produção atual e preditivo e visualização das decisões tomadas pelo agente Gerenciador Planta. O TrakSYS foi utilizado para

aquisição e historiamento de dados. As Figuras 4.9 e 4.10 apresentam a configuração do historiamento e os dados gerados no banco, respectivamente. Entretanto, a quantidade de dados historiados por meio do TrakSYS foi insuficiente para a criação dos modelos de regressão linear citados na seção 4.3.1. Contudo, o molde do conjunto de dados gerados foi útil para a criação de dados simulados em uma quantidade correspondente a dois anos de historiamento.

Os códigos de geração de dados simulados, bem como da criação dos modelos, estão disponíveis no endereço do repositório no Apêndice B, nos arquivos “gerarDadosDisponibilidade.ipynb”, “gerarDadosPedidos.ipynb”, “construcaoModelo1.ipynb” e “construcaoModelo2.ipynb”. Os conjuntos de dados gerados, assim como os modelos exportados também estão disponíveis neste repositório.

Figura 4.9 Configuração de historiamento de variáveis (estado de estações)



Factory > Administration > Services > WIN-N8A3SS6ENHG [ Logic ]  
**Logic Service Tags [ WIN-N8A3SS6ENHG : WIN-N8A3SS6ENHG ]**

Filters	Tags																																																																																																						
<b>Tag Filter 1</b> <input type="text" value="Tipo"/> <input type="button" value="Full or Partial Tag Name"/>	<table border="1"> <thead> <tr> <th>Name</th> <th>Value</th> <th>Quality</th> <th>Type</th> <th>Updated</th> <th>Actions</th> </tr> </thead> <tbody> <tr> <td>TransporteEmergencia_Geral_Accionado</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:04 [ 29.1 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Estação_Disponivel</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:04 [ 29.1 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Estação_Sem_Falhas</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:04 [ 29.1 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Estera_Peralizada_Emergencia</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:04 [ 29.1 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Falha_Inversor</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:04 [ 29.1 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Inspeção_Devidido</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Inspecao_Entregue</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Inter_Enviado</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Limpeza_Devidido</td> <td>23</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Limpeza_Entregue</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte ID_Peca_Saida_Entregue</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Insp_ Entregou_Peca</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Insp_Removeu_Peca</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Inter_ Entregou_Peca</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Limp_ Entregou_Peca</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> <tr> <td>Transporte Limp_Removeu_Peca</td> <td>0</td> <td>192</td> <td>OPC UA</td> <td>fev 23 01:18:05 [ 29.0 Minute(s) Ago ]</td> <td>[ Edit Value ] [ Configuration ]</td> </tr> </tbody> </table>	Name	Value	Quality	Type	Updated	Actions	TransporteEmergencia_Geral_Accionado	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Estação_Disponivel	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Estação_Sem_Falhas	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Estera_Peralizada_Emergencia	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Falha_Inversor	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Inspeção_Devidido	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Inspecao_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Inter_Enviado	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Limpeza_Devidido	23	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Limpeza_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte ID_Peca_Saida_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Insp_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Insp_Removeu_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Inter_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Limp_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]	Transporte Limp_Removeu_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]
Name	Value	Quality	Type	Updated	Actions																																																																																																		
TransporteEmergencia_Geral_Accionado	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Estação_Disponivel	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Estação_Sem_Falhas	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Estera_Peralizada_Emergencia	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Falha_Inversor	0	192	OPC UA	fev 23 01:18:04 [ 29.1 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Inspeção_Devidido	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Inspecao_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Inter_Enviado	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Limpeza_Devidido	23	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Limpeza_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte ID_Peca_Saida_Entregue	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Insp_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Insp_Removeu_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Inter_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Limp_ Entregou_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		
Transporte Limp_Removeu_Peca	0	192	OPC UA	fev 23 01:18:05 [ 29.0 Minute(s) Ago ]	[ Edit Value ] [ Configuration ]																																																																																																		

TrakSYS™ | Senal Cimatic | Factory | Home | Pedidos | Estação de Entrada | HistorianTeste | TesteKPI | Importar Pedidos | VaricosEvents | Configuration | Developer | Administration

Factory > Administration > Services > WIN-N8A3SS6ENHG [ Logic ] > Entrada(Intermidiaria) > Intermidiaria\_Boo > Intermidiaria.Estacao\_Disponivel

### WIN-N8A3SS6ENHG [ Logic ]

Intermediaria.Estacao\_Disponivel

ID: 46  
 Type: OPC UA  
 Data Type: Integer  
 Default Value: 1  
 Value: 1  
 Quality: 192  
 Access Name: S7 Siemens  
 Item Name: mas3as"Supervisor\_Variaveis"/Intermediaria\_Boo[T]

Related: [ Tag History Definitions ]

Tag History Definitions

- Intermediaria.Estacao\_Disponivel

Intermediaria.Estacao\_Disponivel

- ID: 8
- Mode: Periodic
- Maximum Interval (Seconds): 60
- Maximum Deviation: 1
- Related: [ Audit Trail ]
- Actions: [ Duplicate ] [ Delete ]

TrakSYS™ | Senal Cimatic | Factory | Home | Pedidos | Estação de Entrada | HistorianTeste | TesteKPI | Importar Pedidos | VaricosEvents | Configuration | Developer

Factory > Administration > Services > WIN-N8A3SS6ENHG [ Logic ] > Entrada(Intermidiaria) > Intermidiaria\_Boo > Intermidiaria.Estacao\_Disponivel

### Intermediaria.Estacao\_Disponivel

Tag History Definition

General

Name: Intermidiaria.Estacao\_Disponivel

Notes:

Mode: Periodic

Maximum Interval (Seconds): 60

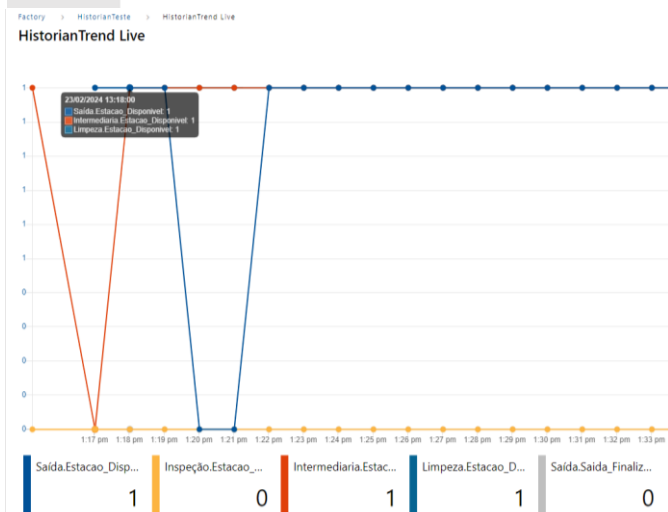
Maximum Deviation: 1

Expiration (Hours): 0

Historian Service: Historian\_PMA

Enabled

[ Apply ] [ Save ] [ Cancel ]



Fonte: Próprio Autor.

Figura 4.10 Conjunto de dados historiados pelo TrakSYS (estado da estação de Saída)

Date Time	Value	Quality
2024-02-29 17:29:45.655 -03:00	0	NULL
2024-02-29 17:30:00.655 -03:00	0	NULL
2024-02-29 17:30:15.655 -03:00	0	NULL
2024-02-29 17:30:30.655 -03:00	0	NULL
2024-02-29 17:30:45.655 -03:00	0	NULL
2024-02-29 17:31:00.655 -03:00	0	NULL
2024-02-29 17:31:15.655 -03:00	0	NULL
2024-02-29 17:31:30.655 -03:00	0	NULL
2024-02-29 17:31:45.655 -03:00	0	NULL
2024-02-29 17:32:00.655 -03:00	0	NULL
2024-02-29 17:32:15.655 -03:00	0	NULL
2024-02-29 17:32:30.655 -03:00	0	NULL
2024-02-29 17:32:45.655 -03:00	0	NULL
2024-02-29 17:33:00.655 -03:00	0	NULL
2024-02-29 17:33:15.655 -03:00	0	NULL
2024-02-29 17:33:30.655 -03:00	0	NULL
2024-02-29 17:33:45.655 -03:00	0	NULL
2024-02-29 17:34:00.655 -03:00	0	NULL
2024-02-29 17:34:15.655 -03:00	0	NULL
2024-02-29 17:34:30.655 -03:00	0	NULL
2024-02-29 17:34:45.655 -03:00	0	NULL
2024-02-29 17:35:00.655 -03:00	0	NULL
2024-02-29 17:35:15.655 -03:00	0	NULL
2024-02-29 17:35:30.655 -03:00	0	NULL
2024-02-29 17:35:45.655 -03:00	0	NULL
2024-02-29 17:36:00.655 -03:00	0	NULL

Fonte: Próprio Autor.

*Este capítulo apresenta discussões relevantes sobre o modelo apresentado, a partir da instância de validação, além de resultados alcançados e outras questões relevantes.*

## **RESULTADOS E DISCUSSÕES**

Com a instância do modelo SAMOM devidamente configurada na PMA, esta foi posta em operação de acordo com o contexto e os critérios descritos na seção 4.2. Na presente seção então, são apresentados resultados que comprovam o alcance dos objetivos da validação proposta.

No que se refere à capacidade em a instância ser distribuída, descentralizada e heterárquica, o cenário de comparação envolve as próprias disposição e configuração das arquiteturas (Figuras 4.5 e 4.6 ou 4.7), como também a operação da PMA sob estas. A comparação é sintetizada no Quadro 5.1. Sob a arquitetura convencional, os dispositivos físicos são controlados pelos CLP 1 a 5, que por sua vez são controlados pelo CLP 6, o qual responde ao controle do SCADA. As decisões são tomadas por um profissional e configuradas na interface do SCADA. Por outro lado, sob a SOA da SAMOM, apesar de ainda serem centralizados no CLP 6, os dispositivos físicos, que respondem aos CLP 1 a 5, possuem representações digitais (agentes) não sujeitos ao controle do CLP 6. Estes agentes operam em conjunto, com coordenação e cooperação previamente configuradas, por meio da troca de informações, principalmente preditivas. Eles também interagem disponibilizando informações ao MES (Figura 5.4) para que haja um acompanhamento da operação do sistema.

Contudo, os agentes não dependem exclusivamente de qualquer sistema. As Figuras 5.1 e 5.2 trazem uma captura das telas que mostram a operação do SMA.

Quadro 5.1 Comparação do 1º critério: capacidade em ser distribuída, descentralizada e heterárquica

	Arquitetura Convencional	Instância SAMOM
Controle	DF – CLP local – CLP central – SCADA	DF possuem representações virtuais (agentes) não sujeitos ao CLP 6
Tomada de Decisões	por um profissional (velocidade da esteira ajustada por operador no SCADA)	agentes, em conjunto, com interação com o MES (velocidade da esteira ajustada pelo agente Gerenciador Planta)

Fonte: Próprio Autor.

Figura 5.1 Operação dos agentes Atuadores (ambiente1.py)

```

76
77
78 #Cria roboSaida da classe Atuador
79 roboSaida = Atuador("Robo_Saida", 'ag20')
80 modRoboSaida = joblib.load("construcaoModelo/modRoboSaida.pkl")
81 predicaoSaida = modRoboSaida.predict(DadosEntradaModelo)
82 roboSaida.dadosInformar = {f'Predicao para Disponibilidade da estacao de Saida para o dia {dataDia["Ano"]}{0}\
': predicaoLimpeza[0]}

Agentes Robo_Entrada informa: {'Predicao para Disponibilidade da estacao de Entrada para o dia 2024-6-10': 0.993893197685662}

Predicao do agente Robo_Entrada disponivel na rota da API: {'Predicao para Disponibilidade da estacao de Entrada para o dia 2024-6-10': 0.993893197685662}

Agentes Esteira_Principal informa: {'Predicao para Disponibilidade da estacao de Transporte para o dia 2024-6-10': 0.9922786905061824}

Predicao do agente Esteira_Principal disponivel na rota da API: {'Predicao para Disponibilidade da estacao de Transporte para o dia 2024-6-10': 0.9922786905061824}

Agentes Robo_Inspcao informa: {'Predicao para Disponibilidade da estacao de Inspecao para o dia 2024-6-10': 0.9967092761924194}

Predicao do agente Robo_Inspcao disponivel na rota da API: {'Predicao para Disponibilidade da estacao de Inspecao para o dia 2024-6-10': 0.9967092761924194}

Agentes Limpador_Pneumatico informa: {'Predicao para Disponibilidade da estacao de Limpeza para o dia 2024-6-10': 0.9969187619663407}

Predicao do agente Limpador_Pneumatico disponivel na rota da API: {'Predicao para Disponibilidade da estacao de Limpeza para o dia 2024-6-10': 0.9969187619663407}

Agentes Robo_Saida informa: {'Predicao para Disponibilidade da estacao de Saida para o dia 2024-6-10': 0.9943855834029481}

Predicao do agente Robo_Saida disponivel na rota da API: {'Predicao para Disponibilidade da estacao de Saida para o dia 2024-6-10': 0.9943855834029481}

```

Fonte: Próprio Autor.

O agente Gerenciador Planta é capaz de tomar uma decisão sobre a velocidade da esteira principal. Por outro lado, a programação original dos CLP e do SCADA permanecem inalteradas. O controle da velocidade da esteira ainda pode ser realizado por um operador na interface do SCADA. A alteração que prevalece é a última que foi realizada. Tanto o SMA quanto o operador possuem a informação da velocidade da

esteira e podem alterá-la. Dessa forma, percebe-se que a arquitetura do modelo SAMOM instanciada na PMA é mais distribuída, descentralizada e heterárquica em comparação à arquitetura convencional.

Figura 5.2 Operação dos agentes Sensor Transporte e Gerenciador Planta (ambiente2.py)

```

ambiente2.py
123 decisaoEsteira = {}
124
125 if estadoEsteira == 0:
126     decisaoEsteira = {'A esteira não está em funcionamento': 40}
127
128 elif predicaoTakt[0] == 0:
129     decisaoEsteira = {'A demanda prevista é 0 (s/und). Manter Velocidade da Esteira reduzida durante os próximos 5 minutos': 40}
130
131 elif ritmoPredito >= predicaoTakt[0]:
132     decisaoEsteira = {'Ritmo de produção predito ((ritmoPredito) s/und) atenderá a demanda prevista((predicaoTakt[0]) s/und). Manter Velocidade da Esteira reduzida durante os próximos 5 minutos': 40}
133
134 else:
135     decisaoEsteira = {'Ritmo de produção predito ((ritmoPredito) s/und) não atenderá a demanda prevista((predicaoTakt[0]) s/und). Aumentar Velocidade da Esteira para os próximos 5 minutos': 55}
136     alteracao = True
137
138 gerenciadorPlanta.dadosInformar = {'Takt_Time': {'Predicao para o Takt Time (s/und) da estação de Entrada para o dia (dataDia["Ano"])[0]\
139                                         }-(dataDia["Mes"])[0]\
140                                         }-(dataDia["Dia"])[0]\
141                                         }': predicaoTakt[0]},
142                                     {'Taxa_de_Saida': {'Predicao para o Taxa de Saida (und/h) da PMA para o dia (dataDia["Ano"])[0]\
143                                         }-(dataDia["Mes"])[0]\
144                                         }-(dataDia["Dia"])[0]\
145                                         }': predicaoTaxaSaida[0]},
146                                     'Decisão do Gerenciador_Planta': decisaoEsteira}

```

```

Agentes Gerenciador_Planta informa: {'Takt_Time': {'Predicao para o Takt Time (s/und) da estação de Entrada para o dia 2024-6-10': 198.1887843304409}, 'Taxa_de_Saida': {'Predicao para o Taxa de Saida (und/h) da PMA para o dia 2024-6-10': 13.396058622356094}, 'Decisão do Gerenciador_Planta': {'Ritmo de produção predito (198.178663555565 s/und) não atenderá a demanda prevista(198.1887843304409 s/und). Aumentar Velocidade da Esteira para os próximos 5 minutos': 55}}

Predicao do agente Gerenciador_Planta disponível na rota da API: {'Takt_Time': {'Predicao para o Takt Time (s/und) da estação de Entrada para o dia 2024-6-10': 198.1887843304409}, 'Taxa_de_Saida': {'Predicao para o Taxa de Saida (und/h) da PMA para o dia 2024-6-10': 13.396058622356094}, 'Decisão do Gerenciador_Planta': {'Ritmo de produção predito (198.178663555565 s/und) não atenderá a demanda prevista(198.1887843304409 s/und). Aumentar Velocidade da Esteira para os próximos 5 minutos': 55}}

Requested session timeout to be 3600000ms, got 30000ms instead
Requested session timeout to be 3600000ms, got 30000ms instead

Agente Sensor_Transporte informa - Novo valor de velocidade da esteira: 55

```

Fonte: Próprio Autor.

Para o critério de comparação de habilidades preditivas das arquiteturas, uma síntese é apresentada no Quadro 5.2. Constata-se que, sob a arquitetura convencional, é possível utilizar dados coletados pelo SCADA para analisá-los em painéis com indicadores e gráfico, como mostrado na Figura 5.3. Nestes painéis são apresentadas também algumas projeções futuras. Entretanto, estas não são baseadas em inteligência.

Quadro 5.2 Comparação do 2º critério: habilidades preditivas

	Arquitetura Convencional	Instância SAMOM
Dados e Informações	não provenientes de dados históricos	preditivas provenientes de agentes treinados
Painéis (dashboards)	indicadores e projeções	demonstração no MES - disponibilidade (agentes Atuadores)

Fonte: Próprio Autor.

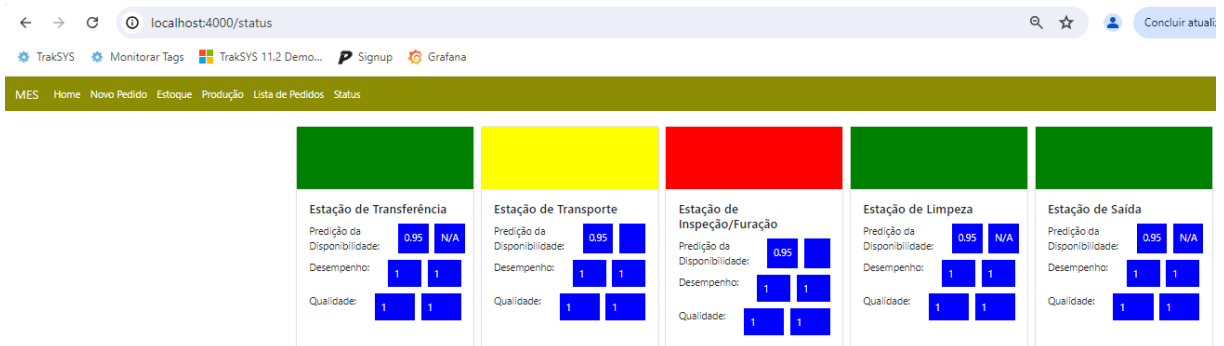
Figura 5.3 Painel de indicadores da operação sob a Arquitetura Convencional



Fonte: Próprio Autor.

Sob o modelo SAMOM, as informações sobre a disponibilidade das estações, provenientes do SMA, são preditivas e demonstradas no MES como apresenta a Figura 5.4. Os agentes são treinados com dados devidamente historiadados e são capazes de realizar tais predições. Estas informações possibilitam a formação de conhecimento sobre a operação da PMA como um todo, o que contribui para uma G&C mais integrada em relação à arquitetura convencional.

Figura 5.4 Captura da tela do AMES em operação junto ao SAMOM



Fonte: Próprio Autor.

Sobre o critério de comparação de habilidades adaptativas, o quadro 5.3 traz uma síntese. Sob a arquitetura convencional, o ritmo de produção está sujeito à programação dos CLP. O operador ou gestor não possui informações suficientes, em intervalo de tempo suficiente, para tomar a decisão de acelerar ou diminuir o ritmo de produção de forma a atender o *takt time*. Por exemplo, ainda que fosse possível calcular o *takt time* em tempo suficiente, seriam necessários dados históricos de demanda dos clientes e o valor encontrado não seria preditivo. Então, a velocidade padrão de 40hz é comumente mantida.

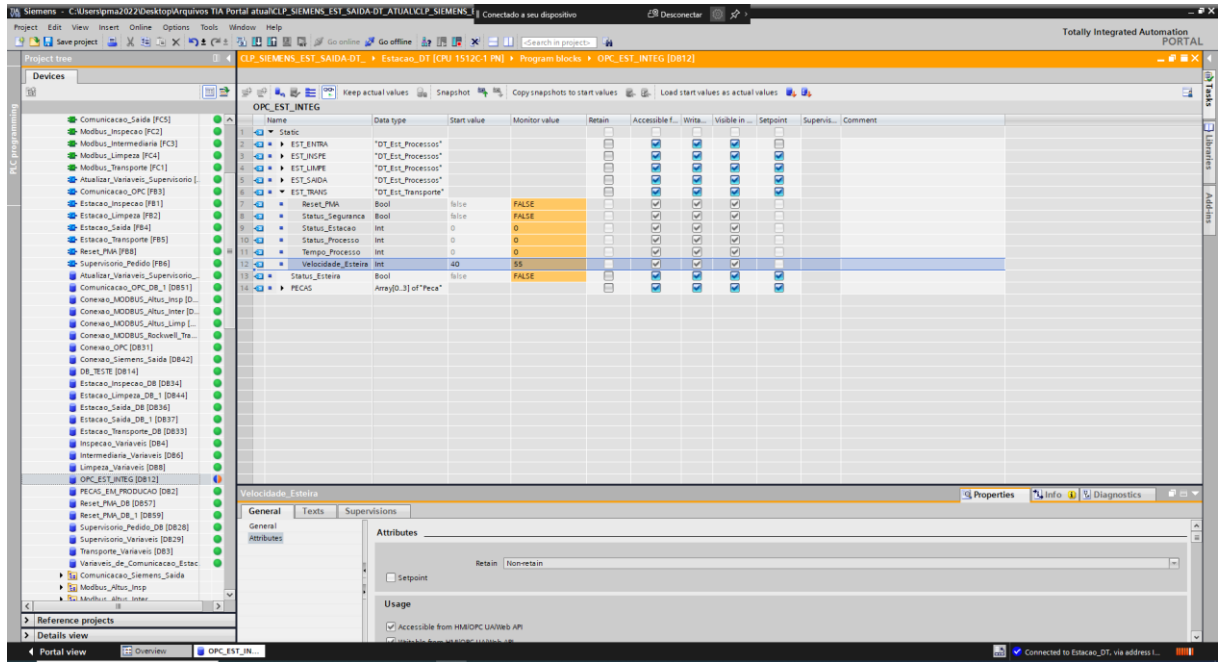
Quadro 5.3 Comparação do 3º critério: habilidades adaptativas

<b>Arquitetura Convencional</b>	<b>Instância SAMOM</b>
<ul style="list-style-type: none"> <li>- sujeito à programação dos CLP;</li> <li>- operador pode ajustar velocidade, mas sem informação preditiva do <i>Takt Time</i>;</li> <li>- 40Hz é mantido como padrão.</li> </ul>	<ul style="list-style-type: none"> <li>- agente Gerenciador recebe disponibilidades, prevê <i>Takt Time</i> e pode decidir entre 40Hz e 55Hz.</li> </ul>

Fonte: Próprio Autor.

Sob o modelo SAMOM, o agente Gerenciador Planta determina um *takt time* predito por seu modelo de regressão linear, baseado no histórico de demanda. Ele determina também um ritmo de produção predito, baseado na disponibilidade das estações. Com isso, ele compara o ritmo predito com o necessário para atender a demanda e decide se reduz para um nível menor (40hz), aumenta para um nível maior (55hz) ou mantém em um dos dois níveis a velocidade da esteira. A Figura 5.5 apresenta a interface de programação do CLP6 onde é demonstrada uma dessas alterações da velocidade.

Figura 5.5 Interface de programação do CLP6 (TIA Portal)



Fonte: Próprio Autor.



*Este capítulo apresenta as conclusões mais relevantes, as contribuições para a academia e a sociedade, bem como elementos para trabalhos futuros que possam dar continuidade a este trabalho de pesquisa.*

## **CONCLUSÃO**

Este trabalho investigou a capacidade de uma arquitetura distribuída e orientada a serviços fornecer habilidades preditivas e adaptativas à G&C de sistemas de manufatura discreta e em batelada. Tanto a arquitetura, quanto o próprio sistema de manufatura são apoiados por tecnologias, conceitos e oportunidades da I4.0. A motivação do objetivo do trabalho baseia-se na emergência de novos modelos de negócio na própria transição para a I4.0. A flexibilidade para a oferta de produtos personalizados é a característica, requerida por tais modelos, mais pertinente à tal motivação. O presente trabalho alcançou objetivo em questão a partir da realização de um estudo, que por sua vez gerou um modelo que propõe a referida arquitetura. Este modelo foi apresentado, validado e discutido.

A parte inicial do estudo definiu o escopo necessário ao trabalho no que se refere principalmente à G&C, operação, e habilidades preditivas e adaptativas de um sistema de manufatura. Tal definição foi fundamentada em discussões com a literatura relacionadas a conceitos próximo aos referidos. Esta literatura foi levantada e analisada por um método sustentado por uma metodologia de pesquisa científica que elencou referências relevantes.

As referências incluídas, bem como a metodologia seguida, também auxiliaram na proposição e na validação do modelo, o SAMOM. Na sua apresentação, foram descritas a arquitetura, regras e características que o compõem. Com isso foi possível compreender como tal modelo pode estabelecer aos sistemas de manufatura as habilidades propostas. Entre estas se destacam a inteligência, baseada em dados e IA, e a G&C distribuída, baseada em SMA, MES e na SOA.

A validação do SAMOM possibilitou a percepção da importância do contexto específico do sistema de manufatura no qual o modelo é instanciado. Possibilitou também a comparação entre um sistema que opera sob o SAMOM com o mesmo sistema sem tal instância. No caso em questão, numa planta piloto de manufatura 4.0 (a PMA), constatou-se que a arquitetura sob o modelo SAMOM possui maior capacidade de ser distribuída, descentralizada e heterárquica. Tal fato é percebido pela inclusão dos agentes como representações digitais dos dispositivos físicos, o que possibilita maior troca de informações de G&C entre cada componente do sistema. Esta troca, por sua vez, gera um conhecimento maior sobre o sistema como um todo, que permite a tomada de decisão e o controle mais integrados.

Foi possível também constatar na validação a capacidade do SAMOM em atribuir habilidades preditivas e adaptativas na PMA. A partir de dados historizados e sintetizados, foram criados modelos de ML que foram utilizados pelos agentes para realizar previsões da disponibilidade de cada da planta, da taxa de produção da planta, bem como do ritmo de demanda do mercado. O SMA foi configurado então para realizar autonomamente a adaptação de manter a velocidade da esteira em um estado menor (40hz da rotação do seu motor) ou aumentar para um estado maior (55hz), a partir da análise da compatibilidade entre a disponibilidade, a taxa de produção e o ritmo de demanda preditos.

Com cada objetivo do trabalho alcançado, conclui-se que este pode ser posto como um alicerce para implementação de arquiteturas compatíveis com sistemas de manufatura 4.0, para novas abordagens da aplicação de tecnologias e conceitos específicos na G&C de sistemas de manufatura e para abordagens de contextos específicos de G&C relacionados a tomada de decisão.

## 6.1 OPORTUNIDADES PARA TRABALHOS FUTUROS

O presente trabalho dispôs de fatores limitantes e desafios, principalmente de escopo e viabilidade, que ensejam abordagens futuras. O tema abordado é multidisciplinar e, para uma profundidade além da alcançada pelo presente trabalho, requer mais recursos e tempo de pesquisa para as diversas áreas envolvidas (manufatura, negócios, automação industrial, tecnologias da informação e comunicação, inteligência computacional, engenharia de *software* etc.)

No que se refere ao escopo das operações de manufatura e produção abordadas, recomenda-se ampliação do estudo para outras áreas como manutenção e gestão de ativos, gestão da qualidade e logística, distribuição e cadeia de suprimentos. Recomenda-se também ampliação para a manufatura contínua e suas particularidades e complexidades. Tais ampliações podem trazer benefícios nestas áreas, no atual contexto de advento da I4.0.

Quanto à pesquisa de referências, que se estende para análise de conceitos e de trabalhos relacionados, recomenda-se novas etapas de busca. Assim é possível encontrar trabalhos ainda mais recentes, que podem trazer novas abordagens de G&C em sistemas de manufatura 4.0, principalmente no que diz respeito a adoção de tecnologias, proposição de modelos ou arquiteturas e implementações com dados relacionados à sua eficácia. A análise de tais trabalhos pode contribuir com o próprio modelo SAMOM ou gerar outras proposições.

No que se refere às características, critérios e arquitetura do modelo SAMOM, a primeira recomendação é, a partir também de mais trabalhos pesquisados, uma análise mais profunda sobre a comunicação. É interessante para progressão do modelo SAMOM um estudo detalhado de diferentes estruturas protocolos e linguagens que envolvam comunicação entre sistemas. Dessa forma, é possível que sejam identificadas mais vantagens e desvantagens de cada conjunto de soluções, inclusive das propostas no presente trabalho. Assim, pode ser proposta uma solução mais pertinente aos objetivos do modelo SAMOM.

Recomenda-se também, da mesma forma que para a comunicação, uma análise mais profunda quanto a soluções para modelagem e construção do SMA. Novos paradigmas, ferramentas e linguagens podem apresentar benefícios ao SMA proposto pelo modelo SAMOM, em relação às soluções aqui propostas.

Para as questões que envolve a inteligência dos agentes no modelo proposto, recomenda-se mais estudos. No âmbito dos modelos de inferência e aprendizado ou algoritmos complexos, recomenda-se avaliações de diferentes aplicações destes. Isso pode permitir uma melhor compreensão do quão inteligente o SMA pode ser. Como exemplo, pode ser possível avaliar a capacidade de reconfigurabilidade, auto-organização, auto-treinamento ou auto-coordenação dos DIM. No âmbito de paradigmas de IA, a inclusão dos conceitos de ontologias e sistemas holônicos, por exemplo, podem contribuir em elevar a inteligência do modelo SAMOM.

Ainda no que se refere ao modelo proposto, recomenda-se a avaliação e inclusão de métodos para determinar, quantitativa ou qualitativamente, o nível de maturidade, relacionado à I4.0, de um sistema de manufatura. Tal abordagem contribui no entendimento de como deve ser a implementação do SAMOM num sistema de manufatura. Além disso, possibilita também a avaliação quantitativa de quanto o modelo SAMOM pode contribuir para a evolução de um sistema.

Uma outra recomendação de estudos especificamente relacionado ao modelo proposto é a análise do trabalho humano. Recomenda-se uma maior reflexão das implicações, oportunidades e tecnologias envolvidas na implementação do modelo e relacionadas ao trabalhador humano na manufatura. A discussão com referências também é essencial nesse âmbito.

Por fim, uma série de recomendações estão relacionadas à validação do modelo. No presente trabalho foi possível realizar um teste em um determinado escopo. Entretanto, recomenda-se mais testes em diferentes sistemas, com contexto mais amplos, e diferentes componentes. Teste em um sistema real de manufatura, diferente e complementarmente ao aqui feito, numa planta piloto de laboratório, pode trazer novas perspectivas quanto à eficiência da implementação do modelo.

São recomendados também testes em sistemas de manufatura em batelada, como também de diferentes seguimentos industriais, para verificação da compatibilidade e das particularidades inerentes. Além disso, recomenda-se testes que realizem o comparativo entre a implementação do SAMOM e de outras arquiteturas ou modelos em um mesmo sistema de manufatura. Um âmbito de testes também relevante é a avaliação das implicações na economia e nas estratégias de negócio de uma organização.

No âmbito do MES, recomenda-se a avaliação completa de MES de mercado implementados sob a arquitetura do modelo SAMOM, a qual não foi possível ser realizada no presente trabalho. Tal recomendação determina a real compatibilidade dos tradicionais e atuais MES com o SAMOM e as consequentes implicações (por exemplo, a necessidade do estabelecimento de um novo paradigma para desenvolvimento de MES).

## 6.2 PRODUÇÃO ACADÊMICA GERADA

Durante o período de desenvolvimento do presente trabalho, algumas publicações foram geradas e estão elencadas a seguir:

CRUZ, Jhaidan Ribeiro; LEPIKSON, Herman Augusto. Uma Arquitetura para Operação Inteligente de Sistemas de Manufatura com Habilidades Adaptativas. **Proceedings of the 12th Brazilian Congress on Manufacturing Engineering, 2023**. doi://10.26678/ABCM.COBEF2023.COF23-0309.

CRUZ, Jhaidan Ribeiro; LEPIKSON, Herman Augusto. Operating Manufacturing Systems with Intelligent and Adaptive Skills for Custom Products. **Conference on Complex Systems 2023**. Complex System Society, 2023.

## REFERÊNCIAS

ARCIDIACONO, Francesco; SCHUPP, Florian. Investigating the impact of smart manufacturing on firms' operational and financial performance. **Journal of Manufacturing Technology Management**, v. 35, n. 3, p. 458-479, 2024.

ARM, Jakub et al. Automated design and integration of asset administration shells in components of industry 4.0. **Sensors**, v. 21, n. 6, p. 2004, 2021.

BONNEY, Maurice. Reflections on production planning and control (PPC). **Gestão & produção**, v. 7, p. 181-207, 2000.

BUCKHORST, Armin F.; GRAHN, Lea; SCHMITT, Robert H. Decentralized Holonic Control System Model for Line-less Mobile Assembly Systems. **Robotics and Computer-Integrated Manufacturing**, v. 75, p. 102301, 2022.

BUENO, Adauto; GODINHO FILHO, Moacir; FRANK, Alejandro G. Smart production planning and control in the Industry 4.0 context: A systematic literature review. **Computers & Industrial Engineering**, v. 149, p. 106774, 2020.

CHANG, Shing I. Approaches to Implement Statistical Process Control for Manufacturing in Big Data Era. In: **International Manufacturing Science and Engineering Conference**. American Society of Mechanical Engineers, 2017. p. V003T04A009.

CHEN, Xinyu; VOIGT, Tobias. Implementation of the Manufacturing Execution System in the food and beverage industry. **Journal of Food Engineering**, v. 278, p. 109932, 2020.

DING, Han et al. State of AI-based monitoring in smart manufacturing and introduction to focused section. **IEEE/ASME Transactions on Mechatronics**, v. 25, n. 5, p. 2143-2154, 2020.

DOGLIO, Fernando; DOGLIO; CORRIGAN. **REST API Development with Node.js**. Apress, 2018.

FETTERMANN, Diego Castro et al. How does Industry 4.0 contribute to operations management?. **Journal of Industrial and Production Engineering**, v. 35, n. 4, p. 255-268, 2018.

FILIPOV, V.; VASILEV, P. Manufacturing operations management—the smart backbone of Industry 4.0. **Industry 4.0**, v. 1, n. 1, p. 19-24, 2016.

GIL, Antonio Carlos. **Como elaborar projetos de pesquisa**. 4. São Paulo: Atlas, 2002.

HE, Q. Peter; WANG, Jin. Statistical process monitoring as a big data analytics tool for smart manufacturing. **Journal of Process Control**, v. 67, p. 35-43, 2018..

ISA. **ANSI/ISA–88.01–1995**: Batch Control  
Part 1: Models and Terminology. 1<sup>st</sup>. Research Triangle Park, 1995. 95 p.

ISA. **ISA95, Enterprise-Control System Integration**. Research Triangle Park: ISA. [2023?]. Disponível em: <https://www.isa.org/standards-and-publications/isa-standards/isa-standards-committees/isa95>. Acesso em: 29 abr. 2023.

JASKÓ, Szilárd et al. Development of manufacturing execution systems in accordance with Industry 4.0 requirements: A review of standard-and ontology-based methodologies and tools. **Computers in industry**, v. 123, p. 103300, 2020.

KIM, Yun Geon et al. Multi-agent system and reinforcement learning approach for distributed intelligence in a flexible smart manufacturing system. **Journal of Manufacturing Systems**, v. 57, p. 440-450, 2020.

KRUGER, Grant H. et al. Intelligent machine agent architecture for adaptive control optimization of manufacturing processes. **Advanced Engineering Informatics**, v. 25, n. 4, p. 783-796, 2011.

KUSIAK, Andrew. Smart manufacturing. **International Journal of Production Research**, v. 56, n. 1-2, p. 508-517, 2018.

LU, Yuqian; XU, Xun; WANG, Lihui. Smart manufacturing process and system automation—a critical review of the standards and envisioned scenarios. **Journal of Manufacturing Systems**, v. 56, p. 312-325, 2020.

MANUFACTURING ENTERPRISE SOLUTIONS ASSOCIATION. MESA Model: A Framework for Smarter Manufacturing. **MESA International**. 2022. Disponível em: <<https://mesa.org/topics-resources/mesa-model/>>. Acesso em: 21 de julho de 2023.

MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 8. São Paulo: Atlas, 2017.

MCCLELLAN, Michael; WEAVER, Doug. **MESA Model Evolution**: white paper #39. Manufacturing Enterprise Solutions Association. Chandler: MESA, 2011.

MORALES MÉNDEZ, Jonathan David; RODRIGUEZ, Ramon Silva. Total productive maintenance (TPM) as a tool for improving productivity: a case study of application in the bottleneck of an auto-parts machining line. **The International Journal of Advanced Manufacturing Technology**, v. 92, p. 1013-1026, 2017.

MORGAN, Jeff et al. Industry 4.0 smart reconfigurable manufacturing machines. **Journal of Manufacturing Systems**, v. 59, p. 481-506, 2021.

PHUYAL, Sudip; BISTA, Diwakar; BISTA, Rabindra. Challenges, opportunities and future directions of smart manufacturing: a state of art review. **Sustainable Futures**,

v. 2, p. 100023, 2020.

RUNGTUSANATHAM, M. Johnny et al. Survey research in operations management: historical analyses. **Journal of Operations management**, v. 21, n. 4, p. 475-488, 2003.

SCHUH, Günther et al. (Ed.). **Industrie 4.0 maturity index: managing the digital transformation of companies**. acatech – National Academy of Science and Engineering. Munique: Herbert Utz Verlag GmbH, 2017. acatech STUDY series.

SHOJAEINASAB, Ardeshir et al. Intelligent manufacturing execution systems: A systematic review. **Journal of Manufacturing Systems**, v. 62, p. 503-522, 2022.

STEVENSON, William J. **Operations management**. New York: McGraw-Hill/Irwin, 2021.

TANG, Hao et al. CASOA: an architecture for agent-based manufacturing system in the context of industry 4.0. **IEEE Access**, v. 6, p. 12746-12754, 2017.

TRIPATHI, Varun et al. Development of a Data-Driven Decision-Making System Using Lean and Smart Manufacturing Concept in Industry 4.0: A Case Study. **Mathematical Problems in Engineering**, v. 2022, n. 1, p. 3012215, 2022.

USUGA CADAVID, Juan Pablo et al. Machine learning applied in production planning and control: a state-of-the-art in the era of industry 4.0. **Journal of Intelligent Manufacturing**, v. 31, n. 6, p. 1531-1558, 2020.

WANG, Baicun et al. Smart manufacturing and intelligent manufacturing: A comparative review. **Engineering**, v. 7, n. 6, p. 738-757, 2021.

WAN, Jiafu et al. Artificial-intelligence-driven customized manufacturing factory: key technologies, applications, and challenges. **Proceedings of the IEEE**, v. 109, n. 4, p. 377-398, 2020.

WILHELM, Yannick et al. Data Science approaches to quality control in manufacturing: A review of problems, challenges and architecture. In: **Symposium and Summer School on Service-Oriented Computing**. Springer, Cham, 2020. p. 45-65.

ZHUANG, Cunbo; LIU, Jianhua; XIONG, Hui. Digital twin-based smart production management and control framework for the complex product assembly shop-floor. **The international journal of advanced manufacturing technology**, v. 96, n. 1, p. 1149-1163, 2018.



## APÊNDICE A

CÓDIGOS DE OPERAÇÃO DO SMA DO MODELO SAMOM  
INSTANCIADO NA PMA

## A.1 – Sensor.py:

```
from opcua import Client, Server
from flask import request, jsonify
from flask_restful import Resource
import json
import time

#Classe que define os atributos e métodos do Sensor (Conforme diagrama de classes)
class Sensor:
    def __init__(self, nomeAgente, idAgente):
        self.idAgente = idAgente
        self.nomeAgente = nomeAgente
        self.dadosInformar = {}
        self.nodeIdOPCUA = {}
        self.urlFonteOPCUA = {}
        self.nomeServidorOPCUA = f"{nomeAgente}_ServidorOPCUA"

    def lerDados(self, nomeFonte):
        # Cria um cliente OPC UA
        client = Client(self.urlFonteOPCUA[nomeFonte])
        # Conectar ao servidor
        client.connect()
        # Acessa a variável do servidor
        noh = client.get_node(self.nodeIdOPCUA[nomeFonte])
        # Lê o nome da variável
        try:
            nomeVariavel = noh.get_description().Text
        except Exception as e:
            nomeVariavel = self.nodeIdOPCUA[nomeFonte]

        # Lê o valor da variável
        valor = noh.get_value()
        # Desconectar do servidor
        client.disconnect()
        return {nomeVariavel:valor}

    def disponibilizarDadosAPI(self):
        with open(f'{self.nomeAgente}.json', 'w') as file:
            json.dump(self.dadosInformar, file)

        print(f'\nDado a informar, do agente {self.nomeAgente}, disponível na rota da API::
        {self.dadosInformar}\n')

    def disponibilizarDadosOPCUA(self, tempo):
```

```

# Criar o servidor OPC UA
servidorOPCUA = Server()
uri = "http://example.com"
servidorOPCUA.set_endpoint("opc.tcp://localhost:4840")
servidorOPCUA.set_server_name(self.nomeServidorOPCUA)

# Criar um objeto de nó
addspace = servidorOPCUA.register_namespace(uri)
node = servidorOPCUA.get_objects_node()
obj_variaveis = node.add_object(addspace, f"objeto{self.nomeAgente}")
obj_variaveis.add_variable(addspace, list(self.dadosInformar.keys())[0],
list(self.dadosInformar.values())[0])

# Exibir os números de namespace e strings de identificação
print(f"\nNúmero de namespace (ns): {addspace}, String de identificação (s):
{uri}\n")

if tempo != 'continuo':
    servidorOPCUA.start()
    print("Servidor OPC UA iniciado!\n")
    time.sleep(tempo)
    servidorOPCUA.stop()
    print("Servidor OPC UA encerrado!\n")

elif tempo == 'continuo':
    servidorOPCUA.start()
    print("Servidor OPC UA iniciado!\n")

    servidorOPCUA.stop()
    print("Servidor OPC UA encerrado!\n")

#Classe que estabelece uma rota de API para cada Agente
class SensorAPI(Resource):
    def post(self):
        nomeAgente = request.json['nomeAgente']

        with open(f'{nomeAgente}.json', 'r') as file:
            data = json.load(file)
        return jsonify(data)

```

## A.2 – Atuador.py:

```
from flask import request, jsonify
from flask_restful import Resource
from opcua import Server
import requests
import json
import time

#Classe que define os atributos e métodos do OperadorHumano (Conforme diagrama de classes)
class Atuador:
    def __init__(self, nomeAgente, idAgente):
        self.idAgente = idAgente
        self.nomeAgente = nomeAgente
        self.agenteFonte = {}
        self.urlFonte = ""
        self.dadosInformar = {}
        self.nomeServidorOPCUA = f"{nomeAgente}_ServidorOPCUA"

    def receberDados(self):
        response = requests.post(url = self.urlFonte, json = self.agenteFonte)
        if response.status_code == 200:
            # A resposta da API está no formato JSON
            data = response.json()
            return data
        else:
            print('Falha ao obter a variável da API')

    def disponibilizarDadosAPI(self):
        with open(f'{self.nomeAgente}.json', 'w') as file:
            json.dump(self.dadosInformar, file)

        print(f'\nPredicao do agente {self.nomeAgente} disponivel na rota da API:
{self.dadosInformar}\n')

    def disponibilizarDadosOPCUA(self, tempo):

        # Criar o servidor OPC UA
        servidorOPCUA = Server()
        uri = "http://example.com"
        servidorOPCUA.set_endpoint("opc.tcp://localhost:4840")
        servidorOPCUA.set_server_name(self.nomeServidorOPCUA)

        # Criar um objeto de nó
        addspace = servidorOPCUA.register_namespace(uri)
        node = servidorOPCUA.get_objects_node()
        obj_variaveis = node.add_object(addspace, f"objeto{self.nomeAgente}")
        obj_variaveis.add_variable(addspace, list(self.dadosInformar.keys())[0],
list(self.dadosInformar.values())[0])

        # Exibir os números de namespace e strings de identificação
        print(f"Número de namespace (ns): {addspace}, String de identificação (s): {uri}")

        if tempo != 'continuo':
            servidorOPCUA.start()
            print("Servidor OPC UA iniciado!")
            time.sleep(tempo)
            servidorOPCUA.stop()
            print("Servidor OPC UA encerrado!")

        elif tempo == 'continuo':
            servidorOPCUA.start()
            print("Servidor OPC UA iniciado!")

            servidorOPCUA.stop()
            print("Servidor OPC UA encerrado!")
```

```
#Classe que estabelece uma rota de API para cada Agente
class AtuatorAPI(Resource):
    def post(self):
        nomeAgente = request.json['nomeAgente']

        with open(f'{nomeAgente}.json', 'r') as file:
            data = json.load(file)
        return jsonify(data)
```

### A.3 – Gerenciador.py:

```
from flask import request, jsonify
from flask_restful import Resource
from opcua import Client, Server, ua
import requests
import json
import time

#Classe que define os atributos e métodos do Gerenciador (Conforme diagrama de classes)
class Gerenciador:
    def __init__(self, nomeAgente, idAgente):
        self.idAgente = idAgente
        self.nomeAgente = nomeAgente
        self.agentesFonte = {}
        self.urlsFonte = {}
        self.dadosInformar = {}
        self.nomeServidorOPCUA = f"{nomeAgente}_ServidorOPCUA"

    def receberDados(self, agenteFonte, classeFonte):
        response = requests.post(url = self.urlsFonte[classeFonte], json =
self.agentesFonte[agenteFonte])
        if response.status_code == 200:
            # A resposta da API está no formato JSON
            data = response.json()
            return data
        else:
            print('Falha ao obter a variável da API!\n')

    def disponibilizarDadosAPI(self):
        with open(f'{self.nomeAgente}.json', 'w') as file:
            json.dump(self.dadosInformar, file)

        print(f'\nPredicao do agente {self.nomeAgente} disponivel na rota da API:
{self.dadosInformar}\n')

    def disponibilizarDadosOPCUA(self, tempo):

        # Criar o servidor OPC UA
        servidorOPCUA = Server()
        uri = "http://example.com"
        servidorOPCUA.set_endpoint("opc.tcp://localhost:4840")
        servidorOPCUA.set_server_name(self.nomeServidorOPCUA)

        # Criar um objeto de nó
        addspace = servidorOPCUA.register_namespace(uri)
        node = servidorOPCUA.get_objects_node()
        obj_variaveis = node.add_object(addspace, f"objeto{self.nomeAgente}")
        obj_variaveis.add_variable(addspace, list(self.dadosInformar.keys())[0],
list(self.dadosInformar.values())[0])

        # Exibir os números de namespace e strings de identificação
        print(f"Número de namespace (ns): {addspace}, String de identificação (s):
{uri}\n")

        if tempo != 'continuo':
            servidorOPCUA.start()
            print("Servidor OPC UA iniciado!\n")
            time.sleep(tempo)
            servidorOPCUA.stop()
            print("Servidor OPC UA encerrado!\n")

        elif tempo == 'continuo':
            servidorOPCUA.start()
            print("Servidor OPC UA iniciado!\n")
```

```

servidorOPCUA.stop()
print("Servidor OPC UA encerrado!\n")

def alterarVariavel(self, nomeFonte, nodeId, nomeVariavel):
    # Cria um cliente OPC UA
    client = Client(self.urlsFonte[nomeFonte])
    # Conectar ao servidor
    client.connect()
    # Acessa a variável do servidor
    noh = client.get_node(nodeId)
    valorAlterar = list(self.dadosInformar[nomeVariavel].values())[0]
    if type(valorAlterar) == bool:
        client_node_dv = ua.DataValue(ua.Variant(valorAlterar, ua.VariantType.Boolean))
    else:
        client_node_dv = ua.DataValue(ua.Variant(valorAlterar, ua.VariantType.Int16))
    #Altera o valor da variável
    noh.set_value(client_node_dv)
    # Desconectar do servidor
    client.disconnect()

#Classe que estabelece uma rota de API para cada Agente
class GerenciadorAPI(Resource):
    def post(self):
        nomeAgente = request.json['nomeAgente']

        with open(f'{nomeAgente}.json', 'r') as file:
            data = json.load(file)
        return jsonify(data)

```

#### A.4 – APISensor.py:

```
from Sensor import SensorAPI
from flask import Flask
from flask_restful import Api

app = Flask(__name__)
api = Api(app)

api.add_resource(SensorAPI, '/sensor')

if __name__ == '__main__':
    app.run(host='192.168.3.115', port=5003)
```

#### A.5 – APIAtuador.py:

```
from Atuador import AtuadorAPI
from flask import Flask
from flask_restful import Api

app = Flask(__name__)
api = Api(app)

api.add_resource(AtuadorAPI, '/atuador')

if __name__ == '__main__':
    app.run(host='192.168.3.115', port=5001)
```

#### A.6 – APIGerenciador.py:

```
from Gerenciador import GerenciadorAPI
from flask import Flask
from flask_restful import Api

app = Flask(__name__)
api = Api(app)

api.add_resource(GerenciadorAPI, '/gerenciador')

if __name__ == '__main__':
    app.run(host='192.168.3.115', port=5002)
```

## A.7 – ambiente1.py:

```
from Atuator import Atuator
import joblib
from datetime import datetime
import pandas as pd

def obterDataDia():
    dataDia= pd.DataFrame({
        'DateTime':[datetime.today()],
        'Dia': '',
        'DiaDaSemana': '',
        'Mes': '',
        'Ano': '',
        'DiaDaSemana_0': [False],
        'DiaDaSemana_1': [False],
        'DiaDaSemana_2': [False],
        'DiaDaSemana_3': [False],
        'DiaDaSemana_4': [False],
        'DiaDaSemana_5': [False],
        'Mes_1': [False], 'Mes_2': [False], 'Mes_3': [False], 'Mes_4':
[False], 'Mes_5': [False], 'Mes_6': [False],
        'Mes_7': [False], 'Mes_8': [False], 'Mes_9': [False], 'Mes_10':
[False], 'Mes_11': [False], 'Mes_12': [False],
    })

    dataDia['DateTime'] = pd.to_datetime(dataDia['DateTime'])
    dataDia['Dia'] = dataDia['DateTime'].dt.day
    dataDia['DiaDaSemana'] = dataDia['DateTime'].dt.dayofweek
    dataDia['Mes'] = dataDia['DateTime'].dt.month
    dataDia['Ano'] = dataDia['DateTime'].dt.year

    weekday_dummies = pd.get_dummies(dataDia['DiaDaSemana'], prefix='DiaDaSemana')
    month_dummies = pd.get_dummies(dataDia['Mes'], prefix='Mes')
    dataDia.update(weekday_dummies)
    dataDia.update(month_dummies)

    entrada = dataDia[['Dia',
'DiaDaSemana_0','DiaDaSemana_1','DiaDaSemana_2','DiaDaSemana_3', 'DiaDaSemana_4',
'DiaDaSemana_5',\
        'Mes_1', 'Mes_2', 'Mes_3', 'Mes_4', 'Mes_5', 'Mes_6', 'Mes_7', 'Mes_8',
'Mes_9', 'Mes_10', 'Mes_11', 'Mes_12', 'Ano']]

    return entrada, dataDia

DadosEntradaModelo, dataDia = obterDataDia()

#Cria roboEntrada da classe Atuator
roboEntrada = Atuator('Robo_Entrada', 'ag17')
modRoboEntrada = joblib.load('construcaoModelo/modRoboEntrada.pkl')
predicaoEntrada = modRoboEntrada.predict(DadosEntradaModelo)
roboEntrada.dadosInformar = {f'Predicao para Disponibilidade da estação de Entrada para o
dia {dataDia["Ano"]}[0]\
                                }-{dataDia["Mes"]}[0]\
                                }-{dataDia["Dia"]}[0]\
                                }':
predicaoEntrada[0]}

#Cria esteiraPrincipal da classe Atuator
esteiraPrincipal = Atuator('Esteira_Principal', 'ag07')
modEsteira = joblib.load('construcaoModelo/modEsteira.pkl')
predicaoTransporte = modEsteira.predict(DadosEntradaModelo)
esteiraPrincipal.dadosInformar = {f'Predicao para Disponibilidade da estação de Transporte
para o dia {dataDia["Ano"]}[0]\
                                }-{dataDia["Mes"]}[0]\
```



```

}--{dataDia["Dia"][0]\
}':
predicaoTransporte[0]}

#Cria roboInspecao da classe Atuador
roboInspecao = Atuador('Robo_Inspecao', 'ag18')
modRoboInspecao = joblib.load('construcaoModelo/modRoboInspecao.pkl')
predicaoInspecao = modRoboInspecao.predict(DadosEntradaModelo)
roboInspecao.dadosInformar = {f'Predicao para Disponibilidade da estação de Inspeção para o
dia {dataDia["Ano"][0]\
}--{dataDia["Mes"][0]\
}--{dataDia["Dia"][0]\
}':
predicaoInspecao[0]}

#Cria limpadorPneumatico da classe Atuador
limpadorPneumatico = Atuador('Limpador_Pneumatico', 'ag19')
modlimpadorPneumatico = joblib.load('construcaoModelo/modLimpadorPneumatico.pkl')
predicaoLimpeza = modlimpadorPneumatico.predict(DadosEntradaModelo)
limpadorPneumatico.dadosInformar = {f'Predicao para Disponibilidade da estação de Limpeza
para o dia {dataDia["Ano"][0]\
}--{dataDia["Mes"][0]\
}--{dataDia["Dia"][0]\
}':
predicaoLimpeza[0]}

#Cria roboSaida da classe Atuador
roboSaida = Atuador('Robo_Saida', 'ag20')
modRoboSaida = joblib.load('construcaoModelo/modRoboSaida.pkl')
predicaoSaida = modRoboSaida.predict(DadosEntradaModelo)
roboSaida.dadosInformar = {f'Predicao para Disponibilidade da estação de Saída para o dia
{dataDia["Ano"][0]\
}--{dataDia["Mes"][0]\
}--{dataDia["Dia"][0]\
}':
predicaoSaida[0]}

#Opera Agentes
print(f'\nAgente {roboEntrada.nomeAgente} informa: {roboEntrada.dadosInformar}\n')
roboEntrada.disponibilizarDadosAPI()
#roboEntrada.disponibilizarDadosOPCUA(tempo=10)

print(f'\nAgente {esteiraPrincipal.nomeAgente} informa:
{esteiraPrincipal.dadosInformar}\n')
esteiraPrincipal.disponibilizarDadosAPI()
#esteiraPrincipal.disponibilizarDadosOPCUA(tempo=10)

print(f'\nAgente {roboInspecao.nomeAgente} informa: {roboInspecao.dadosInformar}\n')
roboInspecao.disponibilizarDadosAPI()
#roboInspecao.disponibilizarDadosOPCUA(tempo=10)

print(f'\nAgente {limpadorPneumatico.nomeAgente} informa:
{limpadorPneumatico.dadosInformar}\n')
limpadorPneumatico.disponibilizarDadosAPI()
#limpadorPneumatico.disponibilizarDadosOPCUA(tempo=10)

print(f'\nAgente {roboSaida.nomeAgente} informa: {roboSaida.dadosInformar}\n')
roboSaida.disponibilizarDadosAPI()
#roboSaida.disponibilizarDadosOPCUA(tempo=10)

```

## A.8 – ambiente2.py:

```
from Gerenciador import Gerenciador
from Sensor import Sensor
import joblib
from datetime import datetime
import pandas as pd
import time
from inputtimeout import inputtimeout, TimeoutOccurred
import sys

#Função de obter a data do dia
def get_user_input(prompt, timeout):
    try:
        user_input = inputtimeout(prompt=prompt, timeout=timeout)
        if user_input.lower() == 's':
            return True
        else:
            return None
    except TimeoutOccurred:
        return None

def obterDataDia():
    dataDia = pd.DataFrame({
        'DateTime': [datetime.today()],
        'Dia': '',
        'DiaDaSemana': '',
        'Mes': '',
        'Ano': '',
        'DiaDaSemana_0': [False],
        'DiaDaSemana_1': [False],
        'DiaDaSemana_2': [False],
        'DiaDaSemana_3': [False],
        'DiaDaSemana_4': [False],
        'DiaDaSemana_5': [False],
        'Mes_1': [False], 'Mes_2': [False], 'Mes_3': [False], 'Mes_4':
[False], 'Mes_5': [False], 'Mes_6': [False],
'Mes_7': [False], 'Mes_8': [False], 'Mes_9': [False], 'Mes_10':
[False], 'Mes_11': [False], 'Mes_12': [False],
    })

    dataDia['DateTime'] = pd.to_datetime(dataDia['DateTime'])
    dataDia['Dia'] = dataDia['DateTime'].dt.day
    dataDia['DiaDaSemana'] = dataDia['DateTime'].dt.dayofweek
    dataDia['Mes'] = dataDia['DateTime'].dt.month
    dataDia['Ano'] = dataDia['DateTime'].dt.year

    weekday_dummies = pd.get_dummies(dataDia['DiaDaSemana'], prefix='DiaDaSemana')
    month_dummies = pd.get_dummies(dataDia['Mes'], prefix='Mes')
    dataDia.update(weekday_dummies)
    dataDia.update(month_dummies)

    entrada = dataDia[['Dia',
'DiaDaSemana_0', 'DiaDaSemana_1', 'DiaDaSemana_2', 'DiaDaSemana_3', 'DiaDaSemana_4',
'DiaDaSemana_5', \
'Mes_1', 'Mes_2', 'Mes_3', 'Mes_4', 'Mes_5', 'Mes_6', 'Mes_7', 'Mes_8',
'Mes_9', 'Mes_10', 'Mes_11', 'Mes_12', 'Ano']]

    return entrada, dataDia

#Data do dia para entrada do modelo de predição do Takt Time
while True:
    entradaModeloTakt, dataDia = obterDataDia()
    #Cria agentes
```

```

gerenciadorPlanta = Gerenciador('Gerenciador_Planta', 'ag16')
sensorTransporte = Sensor("Sensor_Transporte", 'ag06')

#Importa modelos de ML para o agente GerenciadorPlanta
modGerenciadorTaxaSaida = joblib.load('construcaoModelo/modGerenciadorTaxaSaida.pkl')
modGerenciadorTakt = joblib.load('construcaoModelo/modGerenciadorTakt.pkl')

#Define de quem e de quais endereços o agente GerenciadorPlanta vai receber dados
gerenciadorPlanta.agentesFonte = {'Robo_Entrada': {'nomeAgente': 'Robo_Entrada'},
'Esteira_Principal': {'nomeAgente': 'Esteira_Principal'}, \
                                'Robo_Inspecao': {'nomeAgente': 'Robo_Inspecao'},
'Limpador_Pneumatico': {'nomeAgente': 'Limpador_Pneumatico'}, \
                            'Robo_Saida': {'nomeAgente': 'Robo_Saida'}},
gerenciadorPlanta.urlsFonte = {'Atuador': 'http://192.168.3.115:5001/atuador', 'Sensor':
'http://192.168.3.115:5003/sensor',
                                'CLP6': "opc.tcp://192.168.1.24:4840"}

#Define de quem e de quais endereços o agente SensorTransporte vai receber dados
sensorTransporte.urlFonteOPCUA['Estado_Esteira'] = 'opc.tcp://192.168.1.24:4840'
sensorTransporte.urlFonteOPCUA['Velocidade_Esteira'] = 'opc.tcp://192.168.1.24:4840'
sensorTransporte.nodeIdOPCUA["Estado_Esteira"] =
'ns=3;s="Supervisorio_Variaveis"."Transporte_Boo"[1]'
sensorTransporte.nodeIdOPCUA['Velocidade_Esteira'] =
'ns=3;s="OPC_EST_INTEG"."EST_TRANS"."Velocidade_Esteira"'

#SensorTransporte lê dados e disponibiliza para outros agentes
sensorTransporte.dadosInformar['Estado_Esteira'] =
list(sensorTransporte.lerDados(nomeFonte= 'Estado_Esteira').values())[0]
sensorTransporte.dadosInformar['Velocidade_Esteira'] =
list(sensorTransporte.lerDados(nomeFonte= 'Velocidade_Esteira').values())[0]
sensorTransporte.disponibilizarDadosAPI()

#GerenciadorPlanta recebe a disponibilidade prevista dos outros agentes, pega a data do
dia e os junta num dataframe para ser a entrada da
#predição da taxa de saída
entradaModeloTaxaSaida = {}
entradaModeloTaxaSaida['disponibilidade_entrada'] =
list(gerenciadorPlanta.receberDados(agenteFonte='Robo_Entrada',
classeFonte='Atuador').values())[0]
entradaModeloTaxaSaida['disponibilidade_transporte'] =
list(gerenciadorPlanta.receberDados(agenteFonte='Esteira_Principal',
classeFonte='Atuador').values())[0]
entradaModeloTaxaSaida['disponibilidade_inspecao'] =
list(gerenciadorPlanta.receberDados(agenteFonte='Robo_Inspecao',
classeFonte='Atuador').values())[0]
entradaModeloTaxaSaida['disponibilidade_limpeza'] =
list(gerenciadorPlanta.receberDados(agenteFonte='Limpador_Pneumatico',
classeFonte='Atuador').values())[0]
entradaModeloTaxaSaida['disponibilidade_saida'] =
list(gerenciadorPlanta.receberDados(agenteFonte='Robo_Saida',
classeFonte='Atuador').values())[0]
entradaModeloTaxaSaida = pd.DataFrame(entradaModeloTaxaSaida, index = [0])
entradaModeloTaxaSaida = pd.concat([entradaModeloTakt, entradaModeloTaxaSaida], axis=1)

#GerenciadorPlanta recebe o Estado e a Velocidade da Esteira atuais
estadoEsteira = gerenciadorPlanta.receberDados(agenteFonte='Sensor_Transporte',
classeFonte='Sensor')['Estado_Esteira']
#estadoEsteira = 1
velocidadeEsteira = gerenciadorPlanta.receberDados(agenteFonte='Sensor_Transporte',
classeFonte='Sensor')['Velocidade_Esteira']

#GerenciadorPlanta faz a predição da Taxa de Saída e do Takt Time
predicaoTaxaSaida = modGerenciadorTaxaSaida.predict(entradaModeloTaxaSaida)
if predicaoTaxaSaida[0] < 0:
    predicaoTaxaSaida[0] = 0

```



```
gerenciadorPlanta.alterarVariavel(nomeFonte='CLP6',
nodeId='ns=3;s="OPC_EST_INTEG"."EST_TRANS"."Velocidade_Esteira"', nomeVariavel='Decisão do
Gerenciador_Planta')
gerenciadorPlanta.dadosInformar["Alterar_Velocidade"] = {"Alterar_Velocidade": False}
gerenciadorPlanta.alterarVariavel(nomeFonte='CLP6',
nodeId='ns=3;s="OPC_EST_INTEG"."Status_Esteira"', nomeVariavel='Alterar_Velocidade')

print(f"\nAgente Sensor_Transporte informa - Novo valor de velocidade da esteira:
{list(sensorTransporte.lerDados(nomeFonte= 'Velocidade_Esteira').values())[0]}\n")

time.sleep(15)
```

**Apêndice**

**B**

## **APÊNDICE B**

### **REPOSITÓRIO DOS CÓDIGOS DA INSTÂNCIA DO MODELO SAMOM NA PMA**

Endereço (*link*): [https://github.com/jhaidan42/GEMEO\\_DIGITAL\\_PMA\\_SMA.git](https://github.com/jhaidan42/GEMEO_DIGITAL_PMA_SMA.git)