



**UNIVERSIDADE FEDERAL DE OURO PRETO
ESCOLA DE MINAS
ENGENHARIA DE PRODUÇÃO**



TIAGO DE CASTRO RIBEIRO DOS SANTOS

**DESENVOLVIMENTO DE MATERIAL DIDÁTICO PARA
SIMULAÇÃO COMPUTACIONAL COM ANYLOGIC APLICADO À
ENGENHARIA DE PRODUÇÃO**

**OURO PRETO/MG
2026**

TIAGO DE CASTRO RIBEIRO DOS SANTOS

**DESENVOLVIMENTO DE MATERIAL DIDÁTICO PARA SIMULAÇÃO
COMPUTACIONAL COM ANYLOGIC APLICADO À ENGENHARIA DE
PRODUÇÃO**

Monografia apresentada ao Curso de Graduação em Engenharia de Produção da Universidade Federal de Ouro Preto como requisito parcial para a obtenção do título de Engenheiro de Produção.

Área de concentração: Pesquisa Operacional e Simulação

Orientador: Prof. Davi Neves Pavanelli

**OURO PRETO/MG
2026**



MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DE OURO PRETO
REITORIA
ESCOLA DE MINAS
DEPARTAMENTO DE ENGENHARIA DE PRODUÇÃO,
ADMINISTRAÇÃO E ECON



FOLHA DE APROVAÇÃO

Tiago de Castro Ribeiro dos Santos

Desenvolvimento de Material Didático para Simulação Computacional com AnyLogic Aplicado à Engenharia de Produção

Monografia apresentada ao Curso de Engenharia de Produção da Universidade Federal de Ouro Preto como requisito parcial para obtenção do título de Engenheiro

Aprovada em 05 de Março de 2026

Membros da banca

Me. - Davi Neves Pavanelli - Orientador(a) - Universidade Federal de Ouro Preto
Dr. Paulo Marcos de Barros Monteiro - Universidade Federal de Ouro Preto
Me. Cristiano Luis Turbino de Franca e Silva - Universidade Federal de Ouro Preto

Davi Neves Pavanelli, orientador do trabalho, aprovou a versão final e autorizou seu depósito na Biblioteca Digital de Trabalhos de Conclusão de Curso da UFOP em 05/03/2026



Documento assinado eletronicamente por **Davi Neves Pavanelli, PROFESSOR DE MAGISTERIO SUPERIOR**, em 05/03/2026, às 09:37, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site http://sei.ufop.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0, informando o código verificador **1069669** e o código CRC **7A694AB8**.

RESUMO

O presente Trabalho de Conclusão de Curso propõe o desenvolvimento de um material didático estruturado para o ensino de simulação computacional na Engenharia de Produção, utilizando o software AnyLogic. O estudo apresenta um referencial teórico abrangendo os três principais paradigmas de simulação: Eventos Discretos (DES), Sistemas Dinâmicos (SD) e Modelagem Baseada em Agentes (ABM), demonstrando a necessidade de uma abordagem híbrida para resolver problemas complexos. O trabalho detalha a arquitetura do ambiente de simulação e aplica a metodologia em um estudo de caso de uma Rede Logística Regional, integrando roteamento GIS e controle Kanban. Os resultados demonstram que o material desenvolvido permite ao estudante visualizar a interdependência entre decisões operacionais e estratégicas, além de validar a eficácia do AnyLogic como ferramenta pedagógica para superar a fragmentação do ensino tradicional de simulação.

Palavras-chave: simulação computacional, AnyLogic, material didático, logística, engenharia de produção.

ABSTRACT

This Undergraduate Thesis proposes the development of structured educational material for teaching computational simulation in Production Engineering, using the AnyLogic software. The study presents a theoretical framework covering the three main simulation paradigms: Discrete Event Simulation (DES), System Dynamics (SD), and Agent-Based Modeling (ABM), demonstrating the need for a hybrid approach to solve complex problems. The work details the architecture of the simulation environment and applies the methodology in a case study of a Regional Logistics Network, integrating GIS routing and Kanban control. The results demonstrate that the developed material allows students to visualize the interdependence between operational and strategic decisions, in addition to validating the effectiveness of AnyLogic as a pedagogical tool to overcome the fragmentation of traditional simulation teaching.

Keywords: computational simulation, AnyLogic, teaching material, logistics, production engineering.

LISTA DE FIGURAS

Figura 2.1 Modelo de Simulação de Eventos Discretos (DES)	19
Figura 2.2 – Estrutura de Estoques - Sistemas Dinâmicos (SD)	24
Figura 2.3 – Arquitetura Comportamental de um Agente (Máquina de Estados)	28
Figura 2.4 – Arquitetura de Componentes de um Sistema Ciber-Físico (CPS)	30
Figura 2.5 – Espectro de Abstração na Simulação Híbrida (Multimethod)	33
Figura 3.1 – Interface de desenvolvimento do AnyLogic e suas áreas de trabalho	36
Figura 3.2 – Exemplo de modelagem de processo com blocos.....	38
Figura 3.3 – Representação conceitual do SD.....	39
Figura 3.4 – Exemplo de diagrama de Máquina de Estados no AnyLogic	41
Figura 3.5 – Simulação integrado a Sistemas de Informação Geográfica (GIS)	42
Figura 3.6 – Visualização 3D de alta fidelidade de um terminal de contêineres.....	44
Figura 4.1 – Configuração espacial da rede logística no ambiente GIS	48
Figura 4.2 – Diagrama de processos internos do Atacadista e lógica sequencial.....	49
Figura 4.3 – Execução da simulação com fluxo de transporte em tempo real	51

SUMÁRIO

1 INTRODUÇÃO	14
1.1 Considerações Iniciais	14
1.2 Justificativa do Trabalho	15
1.3 Objetivos	16
1.3.1 Objetivo Geral	16
1.3.2 Objetivos Específicos	16
1.4 Organização do Trabalho	17
2 REFERENCIAL TEÓRICO	17
2.1 Eventos Discretos na Engenharia de Produção	18
2.1.1 Conceitos Fundamentais e Definição	18
2.1.2 Aplicações e Relevância na Engenharia de Produção	19
2.1.3 Ligação com a Formação do Engenheiro de Produção	21
2.2 Sistemas Dinâmicos na Engenharia de Produção	22
2.2.1 Conceitos Fundamentais e Definição	22
2.2.2 Aplicações e Relevância na Engenharia de Produção	24
2.2.3 Ligação com a Formação do Engenheiro de Produção	26
2.3 Modelagem Baseada em Agentes na Engenharia de Produção	27
2.3.1 Conceitos Fundamentais e Definição	27
2.3.2 Aplicações e Relevância na Engenharia de Produção	29
2.3.3 Ligação com a Formação do Engenheiro de Produção	31
2.4 Simulação Híbrida e Multimethod Modeling	32
3 O AMBIENTE PARA AS SIMULAÇÕES (MATERIAIS E MÉTODOS)	34
3.1 A Interface do AnyLogic e a Fundamentação Orientada a Objetos	35
3.2 Módulos para a Simulação de Eventos Discretos	37
3.3 Módulos para a Simulação de Sistemas Dinâmicos	38
3.4 Módulos para a Modelagem Baseada em Agentes	40
3.5 Simulação Híbrida (Multimethod Modeling)	42
3.6 Dados, Conectividade e Experimentos de Otimização.....	43

3.7 Bibliotecas Específicas de Domínio (Material Handling e Fluid).....	43
3.8 Entrega via Cloud Computing.....	45
3.9 Integração com Inteligência Artificial e Python.....	45
4 ESTUDO DE CASO: MODELAGEM ESTOCÁSTICA DE UMA REDE LOGÍSTICA REGIONAL COM ROTEAMENTO GIS E CONTROLE KANBAN	46
4.1 Considerações Iniciais e Objetivos do Estudo	46
4.2 Caracterização do Sistema e Parâmetros de Entrada	46
4.2.1 Estrutura dos Agentes	47
4.2.2 Parâmetros da Frota e Demanda	47
4.3 Metodologia de Implementação Híbrida	47
4.3.1 Roteamento em Malha Viária Real (GIS)	48
4.3.2 Modelagem de Processos Internos (DES)	48
4.3.3 Lógica de Controle: Kanban Eletrônico	49
4.4 Verificação, Validação e Análise de Resultados	50
4.4.1 Análise do Roteamento Dinâmico	50
4.4.2 Desempenho da Política Kanban e Gargalos	51
4.5 Discussão e Contribuições Didáticas	52
4.5.1 Limitações do Modelo	52
4.5.2 Valor Didático do Artefato	52
5 CONCLUSÃO.....	53
5.1 Síntese e Cumprimento dos Objetivos	53
5.2 Limitações do Estudo	54
5.3 Recomendações para Trabalhos Futuros	54
5.4 Considerações Finais	55
6 REFERÊNCIAS	55

1 INTRODUÇÃO

1.1 Considerações Iniciais

A Engenharia enfrenta constantemente o desafio de analisar, projetar e aprimorar sistemas cada vez mais complexos, que englobam processos produtivos, logísticos e de serviços. Esses sistemas caracterizam-se por apresentar elevada variabilidade, incerteza e interdependência entre seus componentes, o que torna a tomada de decisão baseada exclusivamente em métodos analíticos, muitas vezes, insuficiente ou inviável.

Nesse contexto, a simulação computacional destaca-se como uma ferramenta estratégica de apoio à decisão, possibilitando a representação de sistemas reais em um ambiente virtual, no qual diferentes cenários podem ser testados de forma segura, ágil e econômica. Por meio da simulação, é possível examinar o comportamento dinâmico dos sistemas ao longo do tempo, avaliar políticas de estoque, planejamento da produção, regras de atendimento, dimensionamento de recursos, arranjo físico (layout) e alocação de capacidades, reduzindo substancialmente os riscos e custos associados à implementação de mudanças no ambiente real.

- No ensino de Engenharia, a simulação assume papel essencial na consolidação dos conceitos teóricos abordados em disciplinas como Planejamento e Controle da Produção (PCP), Logística, Pesquisa Operacional, Sistemas Produtivos e Engenharia de Processos. Tradicionalmente, softwares como Arena — amplamente empregado na modelagem por DES, adequado para representar sistemas de filas, linhas de produção e processos logísticos;
- Minitab — focado em análise estatística e controle de qualidade, sendo utilizado para análise de dados e experimentos;
- Simulink — voltado principalmente para simulação de sistemas dinâmicos contínuos, com aplicação em controle e automação.

Contudo, observa-se que muitos cursos de graduação em Engenharia acabam restringindo o ensino de simulação a apenas uma dessas ferramentas, limitando a formação e a visão sistêmica do estudante. Nesse cenário, o software

AnyLogic emerge como uma alternativa completa e integradora, pois reúne, em uma única plataforma, os três principais paradigmas de simulação:

eventos discretos (Discrete Event Simulation – DES), *Sistemas Dinâmicos* (System Dynamics – SD) e *modelagem baseada em agentes* (Agent-Based Modeling – ABM). Essa versatilidade permite a construção de modelos híbridos e multimétodo, tornando o AnyLogic especialmente adequado para aplicações realistas no contexto da Engenharia de Produção e áreas correlatas.

1.2 Justificativa do Trabalho

A formação de um engenheiro contemporâneo exige a integração entre teoria e prática, sendo a simulação computacional uma competência estratégica indispensável para a análise e melhoria de sistemas reais em ambientes industriais e organizacionais. Restringir-se ao domínio de apenas uma ferramenta computacional constitui um equívoco pedagógico e profissional, uma vez que cada software apresenta potencialidades e limitações específicas. Conhecer e dominar uma plataforma mais abrangente, como o AnyLogic, representa um diferencial acadêmico significativo e amplia as possibilidades de atuação profissional do futuro engenheiro.

Apesar de seu elevado potencial e crescente adoção no mercado internacional, ainda existe carência de materiais didáticos estruturados em língua portuguesa que abordem o AnyLogic de forma contextualizada, partindo de problemas clássicos de engenharia e conduzindo o estudante de maneira gradual e progressiva. Grande parte dos recursos disponíveis encontra-se em inglês, o que dificulta o acesso de muitos estudantes brasileiros e latino-americanos, criando uma barreira adicional ao processo de aprendizagem.

Dessa forma, este trabalho justifica-se pela necessidade de complementar o conjunto de ferramentas utilizadas nos cursos de graduação em Engenharia, oferecendo um material didático robusto, acessível e fundamentado em casos práticos, que facilite o ensino do AnyLogic e contribua efetivamente para sua maior difusão e consolidação no meio acadêmico brasileiro.

Além da barreira do idioma, o acesso corporativo à ferramenta apresenta um alto custo de investimento. Estima-se no mercado de softwares de simulação que as licenças da

versão *Professional*, voltadas para aplicações industriais e empresariais de larga escala, possuam valores estritamente sob consulta que frequentemente giram em torno de 15 a 20 mil dólares por licença. Diante desse cenário corporativo restritivo, o presente trabalho ganha ainda mais relevância por focar na democratização da tecnologia: todo o material didático e o estudo de caso foram estruturados para rodar perfeitamente na versão *Personal Learning Edition* (PLE). Por ser gratuita, essa versão permite que instituições de ensino e estudantes universitários aprendam e reproduzam simulações logísticas complexas sem qualquer ônus financeiro.

1.3 Objetivos

1.3.1 Objetivo Geral

Desenvolver um material didático aplicado e integrador, fundamentado em problemas reais da Engenharia de Produção, com o objetivo de capacitar os estudantes na modelagem e análise de sistemas complexos através da simulação multimétodo no software AnyLogic, promovendo a integração entre a teoria acadêmica e a prática da tomada de decisão sistêmica.

1.3.2 Objetivos Específicos

- Apresentar o referencial teórico essencial para a elaboração de simulações aplicadas à engenharia, abordando os principais paradigmas de modelagem: eventos discretos, Sistemas Dinâmicos e modelagem baseada em agentes;
- Apresentar o ambiente de simulação do software AnyLogic, destacando seus principais recursos, funcionalidades e interfaces de uso;
- Realizar estudos de caso elucidativos, baseados em problemas clássicos da engenharia de produção, demonstrando a aplicação prática e sistemática do AnyLogic.

1.4 Organização do Trabalho

Este trabalho está estruturado em seis capítulos, organizados da seguinte forma:

O capítulo 1 apresenta a introdução ao tema, contextualizando a importância da simulação computacional para a engenharia, a justificativa do trabalho, os objetivos gerais e específicos, e a organização estrutural do documento;

O capítulo 2 apresenta o referencial teórico, abordando os principais paradigmas de simulação: eventos discretos, sistemas dinâmicos e modelagem baseada em agentes, além de conceitos fundamentais de modelagem e análise de sistemas;

O capítulo 3 apresenta o desenvolvimento do material didático no AnyLogic, incluindo a descrição do ambiente de simulação, os estudos de caso desenvolvidos e a documentação dos modelos criados;

O capítulo 4 apresenta os resultados obtidos nos estudos de caso, a análise crítica dos modelos desenvolvidos e a discussão sobre as contribuições do material didático para o ensino de simulação;

No último capítulo serão apresentadas as conclusões do trabalho, sintetizando os principais resultados alcançados e propondo sugestões para trabalhos futuros na área de simulação aplicada ao ensino de engenharia.

2 REFERENCIAL TEÓRICO

A simulação computacional define-se como o processo de projetar um modelo lógico-matemático de um sistema real e conduzir experimentos virtuais com o objetivo de compreender seu comportamento e avaliar estratégias. Na Engenharia de Produção, essa competência é estratégica para a otimização de sistemas produtivos e logísticos, permitindo a antecipação de cenários e a mitigação de riscos operacionais.

A seguir, são discutidos os paradigmas de DES, SD e ABM, que constituem o alicerce teórico para o desenvolvimento de soluções analíticas avançadas.

2.1 Eventos Discretos na Engenharia de Produção

2.1.1 Conceitos Fundamentais e Definição

A Simulação de Eventos Discretos (*Discrete Event Simulation – DES*) é o paradigma de modelagem predominante para sistemas operacionais, no qual o estado do sistema evolui exclusivamente através de transições instantâneas que ocorrem em pontos discretos no tempo, denominados eventos [LAW, 2015].

Diferentemente da simulação contínua — regida por equações diferenciais onde as variáveis se alteram ininterruptamente —, a DES fundamenta-se na premissa de que o sistema permanece estático entre dois eventos consecutivos. As mudanças de estado (como a alteração do nível de estoque ou a ocupação de uma máquina) são disparadas apenas pela ocorrência de eventos pontuais, garantindo eficiência computacional ao focar o processamento apenas nos momentos de interesse.

Os componentes estruturais que definem a arquitetura de um modelo de eventos discretos são: [BANKS et al., 2010]:

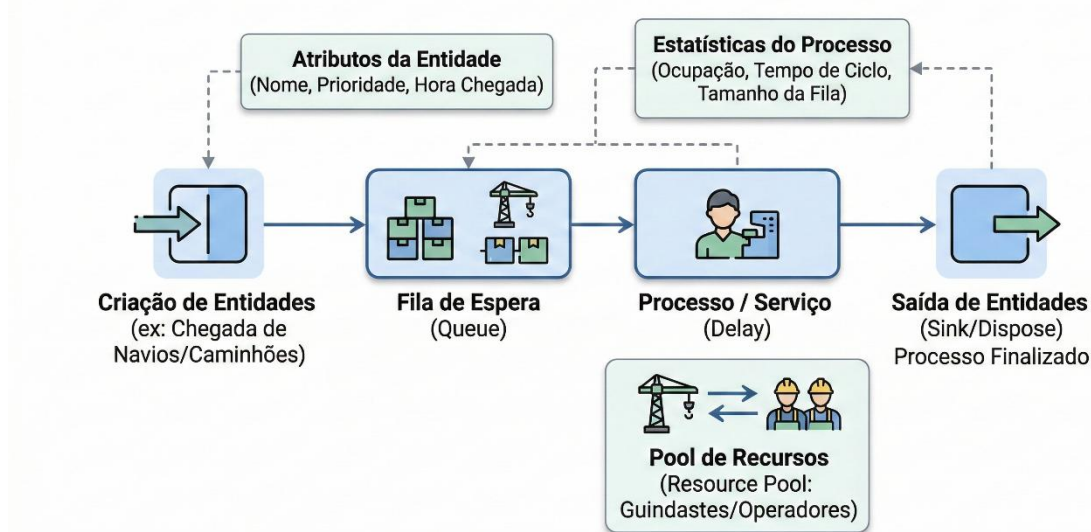
- Entidades: Objetos dinâmicos que transitam pelo sistema demandando recursos. Representam o fluxo de valor (peças, clientes, ordens de produção) e possuem atributos individuais (ex: tempo de ciclo, prioridade, tipo de defeito) que determinam seu comportamento dentro do modelo;
- Recursos: Elementos com capacidade finita responsáveis por processar as entidades. Incluem máquinas, operadores e equipamentos de movimentação. A interação entre a demanda das entidades e a disponibilidade limitada dos recursos é o que gera as filas e gargalos do sistema;
- Filas: Buffers ou áreas de acumulação onde as entidades aguardam quando os recursos necessários estão indisponíveis. O comportamento da fila é regido por uma disciplina de atendimento (como *FIFO*, *LIFO* ou Prioridade), sendo um indicador crítico de balanceamento de linha;
- Eventos: Gatilhos lógicos que alteram instantaneamente as variáveis de estado do sistema. Exemplos típicos incluem "chegada de entidade", "fim de serviço" ou "falha de equipamento";

- **Relógio de simulação:** Variável global que rastreia o tempo virtual do modelo. Utiliza o mecanismo de "avanço para o próximo evento" (*next-event time advance*), saltando os intervalos de inatividade para otimizar a execução.

A lógica de execução de um modelo DES baseia-se no gerenciamento de uma Lista de Eventos Futuros (FEL – *Future Event List*). O motor de simulação processa sequencialmente o evento iminente na lista, atualiza o relógio para o instante desse evento e altera o estado do sistema, agendando novos eventos conforme a lógica do processo. Essa abordagem é ideal para analisar sistemas estocásticos onde a ocorrência de eventos é irregular e espaçada no tempo. [KELTON; SADOWSKI; ZUPICK, 2014].

Conforme ilustrado na Figura 1, essa dinâmica orquestra o fluxo contínuo onde entidades geram demandas, formam filas por limitações de capacidade e ocupam recursos antes de saírem do sistema.

Figura 2.1: Arquitetura Genérica de um Modelo de Simulação de Eventos Discretos (DES)



Fonte: Elaboração do autor (2026).

2.1.2 Aplicações e Relevância na Engenharia de Produção

Para o Engenheiro de Produção, a DES transcende a condição de ferramenta computacional para se tornar uma competência analítica central. Ela representa o paradigma mais aderente à realidade da manufatura e logística, pois é capaz de

incorporar a variabilidade estocástica (falhas de máquinas, absenteísmo, flutuação de demanda) e as interdependências sistêmicas que planilhas estáticas falham em capturar.

Suas principais aplicações estratégicas incluem:

- Otimização de Sistemas de Manufatura: A DES permite "estressar" virtualmente a linha de produção para localizar gargalos dinâmicos (que mudam de lugar) e otimizar o nível de *WIP* (*Work in Process*). É utilizada para validar o balanceamento de linhas, definir políticas de *setup* e manutenção, e testar o impacto da introdução de novos produtos. Estudos clássicos validam que a simulação reduz o *lead time* e maximiza o *throughput* sem a necessidade de paradas físicas para testes [BUZACOTT; SHANTHIKUMAR, 1993];
- Logística Portuária e Cadeias de Suprimentos: Em ambientes de alto custo de capital, como terminais portuários e CDs, a DES é vital para dimensionar a capacidade infraestrutural frente à demanda incerta. A modelagem confronta a capacidade estática (berços, portêineres) com a chegada aleatória de navios e caminhões, gerando indicadores críticos como a taxa de ocupação e tempos de fila. Essa análise é pré-requisito para mitigar custos severos de *demurrage* e fundamentar decisões de investimento (CAPEX) [DRAGOVIC et al., 2017];
- Projeto de Layout e Intralogística: Integrada aos conceitos de *Lean Manufacturing*, a simulação auxilia na validação de arranjos físicos, testando fluxos celulares versus funcionais. Ela permite quantificar o impacto de diferentes layouts sobre a movimentação de materiais e a utilização de empilhadeiras ou AGVs (*Automated Guided Vehicles*), minimizando desperdícios de transporte antes da instalação física [TOMPKINS et al., 2010];
- Engenharia de Serviços e Filas: Em sistemas de serviços (hospitais, *call centers*, aeroportos), o foco desloca-se para o *trade-off* entre custo operacional e nível de serviço. A DES é empregada para dimensionar equipes e recursos a fim de manter o tempo de espera dentro de limites aceitáveis, lidando com picos de demanda. A capacidade de prever a formação de filas torna a ferramenta indispensável para a gestão da experiência do cliente [GÜNAL; PIDD, 2010].

A relevância da DES na formação do engenheiro reside na possibilidade de experimentação livre de riscos (*risk-free environment*). Ela permite que o profissional

antecipe falhas de projeto e valide cenários "E-se" (What-if scenarios), consolidando-se como um instrumento de suporte à decisão robusto para a melhoria contínua e inovação em processos complexos [ROBINSON, 2014].

2.1.3 Ligação com a Formação do Engenheiro de Produção

A apropriação dos princípios da DES constitui uma competência estruturante para o engenheiro de produção, transcendendo o mero uso de software para desenvolver uma visão sistêmica robusta. O domínio da lógica DES obriga o estudante a abandonar o pensamento linear e fragmentado, capacitando-o a compreender como interações locais e aleatórias impactam o desempenho global do sistema. Essa habilidade é vital para distinguir entre *eficiência local* (otimizar uma única máquina) e *eficácia global* (garantir o fluxo de valor ao cliente).

Além disso, a DES atua como o elo prático entre as disciplinas de Estatística e a Engenharia Aplicada. Ela proporciona o desenvolvimento da capacidade analítica quantitativa, pois exige que o engenheiro transite desde a coleta e tratamento de dados brutos (*Input Modeling*) até a interpretação de resultados estocásticos. O estudante aprende que médias não contam toda a história; é preciso compreender a variabilidade, validar distribuições de probabilidade e realizar testes de hipóteses para sustentar conclusões técnicas [LAW, 2015]. Essa competência é diretamente transferível para áreas como *Data Science*, Controle Estatístico de Processos (CEP) e Pesquisa Operacional.

A modelagem por DES também fomenta o raciocínio baseado em experimentação e gestão de riscos. Ao ensinar que decisões complexas não devem se basear em intuição, a DES introduz a cultura da "prototipagem virtual". O engenheiro aprende a avaliar *trade-offs* (compensações) entre custos e níveis de serviço em um ambiente seguro, fundamentando suas escolhas em evidências quantitativas obtidas da análise sistemática de múltiplos cenários. Essa abordagem científica é um diferencial competitivo crucial em um mercado volátil, onde o custo do erro na implementação real é proibitivo [ROBINSON, 2014].

2.2 Sistemas Dinâmicos na Engenharia de Produção

2.2.1 Conceitos Fundamentais e Definição

Os Sistemas Dinâmicos (System Dynamics – SD) é uma abordagem de modelagem e simulação voltada para o estudo de sistemas complexos e não lineares, onde o comportamento observado ao longo do tempo emerge da estrutura interna de realimentações. Desenvolvida originalmente por Jay Forrester no MIT na década de 1950, sob o nome de Industrial Dynamics, o SD rompe com a visão linear de causa e efeito (Eventos → Reações), propondo uma visão circular e endógena, na qual as políticas e a estrutura do sistema são as principais responsáveis pelos seus sucessos ou falhas, e não apenas fatores exógenos [STERMAN, 2000].

Para o engenheiro, o SD pode ser compreendido como uma metodologia para modelar sistemas de equações diferenciais de alta ordem através de diagramas de fluxo, facilitando a análise de estabilidade, oscilação e crescimento. Os elementos básicos que compõem sua arquitetura são: [FORRESTER, 1961; STERMAN, 2000]:

- Estoques (*Stocks ou Levels*): Representam as variáveis de estado do sistema, ou seja, as acumulações que conferem "memória" e inércia ao processo. Matematicamente, os estoques são integrais dos fluxos ao longo do tempo. Exemplos incluem o nível físico de inventário, o volume de capital em caixa, ou o *know-how* acumulado por uma equipe. Como integrais, eles desacoplam as taxas de entrada e saída, permitindo que estas difiram temporariamente;
- Fluxos (*Flows ou Rates*): Representam as taxas de variação dos estoques, equivalendo matematicamente às derivadas das variáveis de estado. São as ações que alteram o sistema, como ordens de produção/hora, taxa de contratação/mês ou consumo/dia. Enquanto os estoques mostram a situação estática em um "*snapshot*" (foto), os fluxos descrevem a dinâmica de mudança;
- Variáveis Auxiliares e Conversores: Elementos algébricos que processam informações dos estoques e de parâmetros exógenos para controlar os fluxos. Elas

tornam explícitas as regras de decisão e as políticas operacionais. Exemplos incluem o cálculo da "cobertura de estoque desejada" ou a "produtividade média";

- Laços de Realimentação (*Feedback Loops*): A estrutura fundamental de causalidade circular. Existem apenas dois tipos:

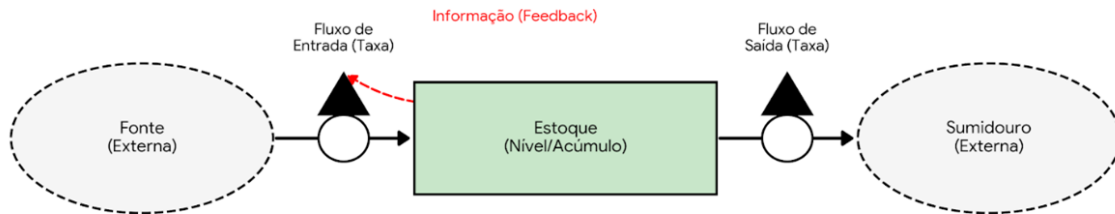
- Reforço (*Positive/Reinforcing*): Geram crescimento exponencial ou colapso (círculos virtuosos ou viciosos), amplificando desvios;

- Equilíbrio (*Negative/Balancing*): Buscam atingir uma meta ou estabilizar o sistema, agindo como forças restauradoras (similar a um controle PID em automação).

- Atrasos (*Delays*): Representam o tempo decorrido entre uma ação e seu resultado perceptível (defasagens temporais). Em engenharia de controle, atrasos em laços de feedback negativo são causas frequentes de instabilidade e oscilação indesejada (ex: o fenômeno do "banho quente" ou o Efeito Chicote na cadeia de suprimentos).

A lógica computacional do SD baseia-se na integração numérica (passo a passo) das equações diferenciais que governam os fluxos. A Figura 2 ilustra essa arquitetura, demonstrando como a informação do nível de estoque retroalimenta as taxas de entrada e saída (fluxos) através de válvulas lógicas. Ao simular essas interações, o modelo revela comportamentos emergentes e contra-intuitivos — como a resistência a políticas de melhoria (*policy resistance*) —, permitindo ao engenheiro projetar estruturas organizacionais mais robustas e resilientes [STERMAN, 2000].

Figura 2.2: Estrutura de Estoque, Fluxos e Realimentação – Sistemas Dinâmicos (SD)



Elaboração do autor (2026), baseado em Sterman (2000).

2.2.2 Aplicações e Relevância na Engenharia de Produção

O SD é a ferramenta por excelência para abordar a complexidade dinâmica — problemas onde a causa e o efeito estão distantes no tempo e no espaço, tornando a intuição linear ineficaz. Diferente da DES (focada no detalhe operacional), o SD é utilizado para o planejamento estratégico e tático, lidando com políticas corporativas e comportamentos agregados.

Suas principais aplicações na Engenharia de Produção incluem:

- **Gestão da Cadeia de Suprimentos e *Bullwhip Effect*:** O fenômeno clássico da área, conhecido como Efeito Chicote, descreve como pequenas flutuações na demanda do consumidor geram oscilações amplificadas a montante da cadeia (fabricantes e fornecedores). O SD permite modelar a estrutura de informação e os atrasos de *lead time* que causam esse descontrole. O modelo atua como um laboratório para testar políticas de compartilhamento de informação (VMI – *Vendor Managed Inventory*) e redução de atrasos, visando estabilizar a cadeia e reduzir o capital de giro imobilizado;
- **Dinâmica de Estoques e Controle de Produção:** Sob a ótica da engenharia de controle, sistemas de estoque são malhas de *feedback* negativo sujeitas a instabilidade. O SD auxilia no ajuste de parâmetros de reposição (ganho e tempo

de ajuste), buscando o equilíbrio ótimo entre custos de manutenção e nível de serviço. Modelos de SD revelam como políticas de "suavização da produção" (*production smoothing*) interagem com a variabilidade da demanda, evitando reações exageradas que desestabilizam o chão de fábrica [TOWILL, 1996];

- Planejamento de Capacidade e Ciclos de Investimento: Decisões de expansão de capacidade (CAPEX) envolvem longos atrasos de construção e implementação. A modelagem por SD captura a dinâmica entre a percepção de demanda, o *backlog* de pedidos e a entrada em operação de novos recursos. Ela é fundamental para evitar os ciclos de "Boom e Bust" (excesso e falta de capacidade), permitindo planejar expansões de forma anticíclica e robusta [LYNEIS, 1999];
- Gestão de Projetos e Ciclos de Retrabalho: Uma aplicação vital na engenharia é a modelagem de grandes projetos (construção civil, desenvolvimento de produtos, software). O SD evidencia o "Ciclo do Retrabalho" e efeitos não lineares como a Lei de Brooks ("adicionar mão de obra a um projeto atrasado o atrasa ainda mais"). O modelo permite simular o impacto da fadiga, da pressão por prazos e da diluição da qualidade, auxiliando gerentes a evitar o colapso do projeto devido a erros ocultos [FORD; STERMAN, 1998];
- Sustentabilidade e Cadeias de Suprimentos de Circuito Fechado: O SD é a abordagem padrão para modelar a Economia Circular (*Closed-Loop Supply Chains*). Ele permite analisar a acumulação de produtos no mercado, as taxas de descarte e a eficácia de processos de logística reversa e remanufatura. O modelo integra fluxos físicos e econômicos para avaliar a viabilidade de longo prazo de estratégias ambientais [ANGERHOFER; ANGELIDES, 2000].

A relevância do SD para a Engenharia de Produção reside na sua capacidade de expor as consequências não intencionais de decisões gerenciais. Ao simular as realimentações e atrasos, o engenheiro consegue identificar pontos de alavancagem (*leverage points*) onde pequenas intervenções estruturais geram grandes melhorias de desempenho duradouro, atacando a raiz dos problemas sistêmicos e não apenas seus sintomas. [SENGE, 1990].

2.2.3 Ligação com a Formação do Engenheiro de Produção

A apropriação dos princípios do SD constitui uma competência estratégica de alto nível para o engenheiro de produção, instrumentalizando-o com o pensamento sistêmico (*Systems Thinking*). Essa habilidade transcende a visão analítica tradicional — que busca entender as partes isoladamente — para focar nas interações e na totalidade. Em um ambiente empresarial volátil, o engenheiro capaz de identificar a estrutura de realimentação subjacente aos problemas evita as "soluções de curto prazo" que frequentemente resultam em deterioração do desempenho a longo prazo ("melhora antes de piorar") [SENGE, 1990].

O SD também fomenta a capacidade de análise dinâmica e não linear. Diferente da estatística estática, o SD ensina que os sistemas produtivos possuem inércia e memória. O engenheiro formado com essa mentalidade aprende a diagnosticar padrões de comportamento temporal — como oscilações amortecidas, crescimento em "S" ou colapsos repentinos — e a correlacioná-los com estruturas de atraso (*delays*) e *feedback*. Isso é crucial para evitar intervenções intempestivas que desestabilizam a cadeia de valor [STERMAN, 2000].

Um diferencial crítico desta abordagem é o desenvolvimento da habilidade de modelagem de variáveis intangíveis e formulação de políticas. O SD exige a explicitação matemática de regras de decisão gerenciais e fatores "suaves" (como fadiga, moral da equipe ou percepção de qualidade), frequentemente ignorados em modelos de planilha. Essa competência permite ao engenheiro criar "Simuladores de Voo para Gestão" (*Management Flight Simulators*), testando a robustez de estratégias corporativas em um ambiente virtual seguro antes de comprometer recursos reais.

Por fim, o SD promove uma postura projetual crítica. Ao revelar que a maioria dos problemas organizacionais é endógena (gerada pelas próprias políticas internas e não por "culpados" externos), a metodologia empodera o engenheiro a atuar como um arquiteto organizacional. Ele deixa de ser apenas um solucionador de problemas pontuais para se tornar um projetista de sistemas resilientes, capaz de questionar modelos mentais vigentes e propor inovações estruturais fundamentadas em rigorosa análise causal [FORRESTER, 1961].

2.3 Modelagem Baseada em Agentes na Engenharia de Produção

2.3.1 Conceitos Fundamentais e Definição

A Modelagem Baseada em Agentes (*Agent-Based Modeling* – ABM) representa uma mudança de paradigma na simulação computacional, transitando de uma abordagem focada em processos (como na DES) ou em fluxos agregados (como no SD) para uma perspectiva centrada no indivíduo (*individual-centric*). A ABM é definida como o método computacional que permite simular as ações e interações de agentes autônomos com o objetivo de avaliar seus efeitos no sistema como um todo. [MACAL; NORTH, 2010].

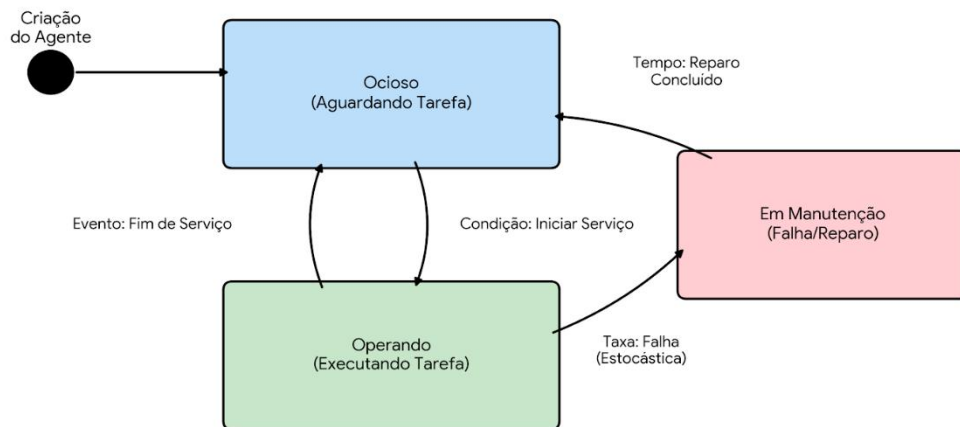
Diferentemente das abordagens *top-down*, onde o comportamento do sistema é regido por equações globais, a ABM adota uma lógica *bottom-up*: as propriedades macroscópicas do sistema (emergência) resultam exclusivamente das micro-decisões descentralizadas de seus componentes.

Os pilares que sustentam a arquitetura de um modelo baseado em agentes são [BONABEAU, 2002; RAILSBACK; GRIMM, 2019]:

- Agentes (Heterogeneidade e Autonomia): São entidades computacionais discretas que possuem identidade própria e comportamentos distintos. Ao contrário das "entidades" passivas da DES, os agentes na ABM são ativos: tomam decisões baseadas em um conjunto de regras internas. Uma característica fundamental é a heterogeneidade: não existe um "agente médio representativo"; cada agente (seja um navio, um consumidor ou uma máquina) pode ter características, histórico e objetivos únicos;
- Ambiente (Topologia e GIS): O meio onde os agentes operam não é apenas um cenário passivo, mas uma topologia que impõe restrições e oportunidades. O ambiente pode ser discreto (celular), contínuo (espaço euclidiano) ou geoespacial (*GIS*), permitindo simular a influência de distâncias reais, relevo e redes logísticas sobre o comportamento dos agentes. O ambiente também pode mediar interações indiretas, fenômeno conhecido como estigmergia;

- **Arquitetura Comportamental (Statecharts):** O "cérebro" do agente. Na engenharia, o comportamento dos agentes é frequentemente modelado através de Máquinas de Estados (*Statecharts*), que definem os estados possíveis do agente (como demonstrado na Figura 2.3: 'Ocioso', 'Operando' ou 'Em Manutenção') e as transições lógicas disparadas por eventos ou condições temporais. As regras podem variar desde simples reações (se-então) até algoritmos complexos de aprendizado e adaptação;

Figura 2.3: Arquitetura Comportamental de um Agente (Máquina de Estados)



Fonte: Elaboração do autor (2026).

- **Interações e Redes:** Os agentes interagem com seus vizinhos e com o ambiente, trocando informações, recursos ou influências. A estrutura dessas interações (quem fala com quem) define a topologia da rede social ou logística do modelo, sendo determinante para a difusão de inovações ou a propagação de falhas;
- **Racionalidade Limitada (*Bounded Rationality*):** Diferente de modelos de otimização global, na ABM os agentes geralmente possuem informações incompletas e visão local. Eles tomam a "melhor decisão possível" com base no que percebem ao seu redor, o que confere maior realismo à simulação de sistemas humanos e de mercado.

Emergência e Complexidade O conceito central da ABM é a emergência: o surgimento de padrões globais complexos, ordenados e muitas vezes contra-intuitivos, que não foram explicitamente programados nas regras individuais. Exemplos clássicos na engenharia incluem a formação espontânea de gargalos de tráfego ("ondas fantasmas"), a auto-organização de cadeias de suprimentos e *tipping points* em dinâmicas de mercado.

A execução de um modelo ABM ocorre através de passos de tempo discretos ou eventos assíncronos, onde o estado de cada agente é atualizado iterativamente. Isso permite ao engenheiro observar não apenas o estado final de equilíbrio, mas a trajetória transiente e a evolução dinâmica do sistema sob diferentes condições iniciais [MACAL; NORTH, 2010].

2.3.2 Aplicações e Relevância na Engenharia de Produção

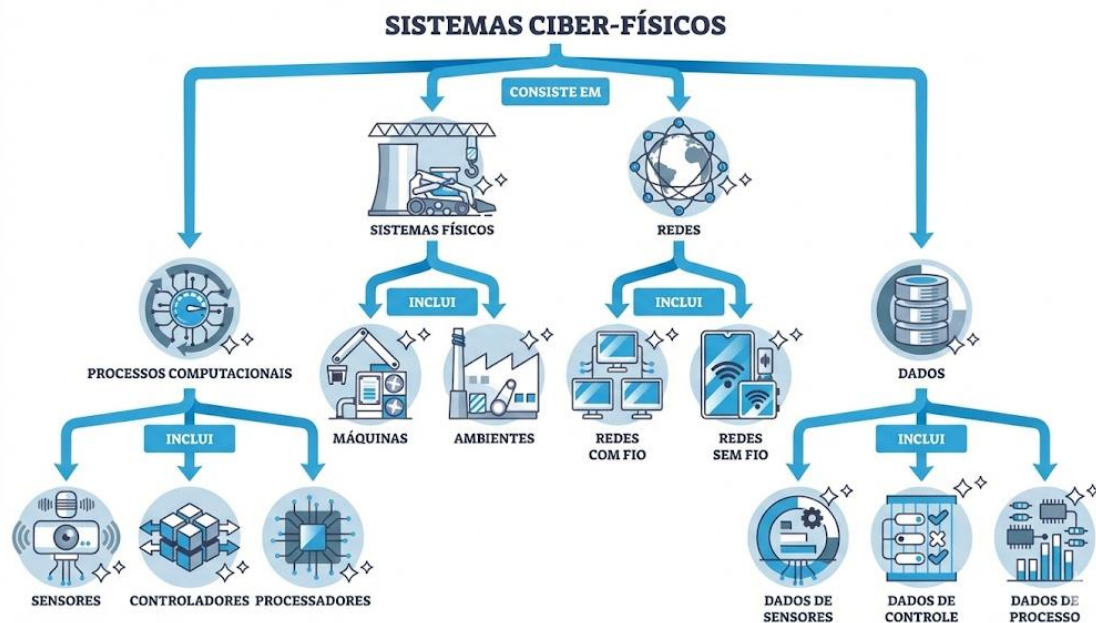
A Modelagem Baseada em Agentes (ABM) transcende as limitações das abordagens estáticas, posicionando-se como a técnica definitiva para analisar sistemas onde a heterogeneidade e a autonomia dos componentes determinam o desempenho global. Ela é especialmente adequada para capturar fenômenos de não-linearidade e adaptação em ambientes dinâmicos.

Suas principais aplicações estratégicas incluem:

- **Logística de Transportes e Ativos Inteligentes:** Diferente da abordagem tradicional de fluxos fluidos, a ABM modela cada ativo (navio, caminhão, contêiner) como um agente único, dotado de atributos específicos (dimensões, tipo de carga, estado de manutenção) e "inteligência" de navegação. Isso permite simular cenários de alta fidelidade onde o desempenho logístico emerge de decisões descentralizadas de roteamento e agendamento. Integrada a Sistemas de Informação Geográfica (GIS), a ABM é vital para validar a capacidade de infraestruturas críticas (portos, ferrovias) frente a gargalos dinâmicos e variabilidade estocástica.
- **Sistemas de Manufatura e Indústria 4.0:** No contexto da *Smart Manufacturing*, sistemas rígidos e centralizados dão lugar a arquiteturas holônicas e descentralizadas. A ABM é a ferramenta natural para modelar Sistemas Ciber-

Físicos (CPS), onde máquinas e produtos "conversam" entre si para negociar prioridades e recursos em tempo real.

Figura 2.4: Arquitetura de Componentes de um Sistema Ciber-Físico (CPS), integrando processos computacionais, sistemas físicos e redes de dados.



Fonte: Elaboração do autor (2026), baseado em Monostori et al. (2016).

Essa abordagem permite testar algoritmos de auto-organização e tolerância a falhas sem paralisar a linha de produção [MONOSTORI et al., 2016];

- **Resiliência em Cadeias de Suprimentos (*Supply Chain Resilience*):** A ABM permite modelar a cadeia não como um fluxo linear, mas como uma rede complexa de agentes (fornecedores, plantas, distribuidores) adaptativos. É fundamental para a análise de riscos e do "Efeito Cascata" (*Ripple Effect*), simulando como uma ruptura local (ex: greve, desastre natural) se propaga pela rede. O modelo revela vulnerabilidades ocultas e permite testar estratégias de mitigação e recuperação [MACAL; NORTH, 2010];
- **Comportamento do Consumidor e Difusão de Inovações:** A ABM simula o mercado como uma rede social de agentes que influenciam uns aos outros. Ao capturar a

irracionalidade e a imitação, a técnica projeta a curva de adoção de novos produtos com maior precisão que métodos estatísticos tradicionais, sendo valiosa para estratégias de *Go-to-Market* e precificação dinâmica [KIESLING et al., 2012];

- Fluxos de Pedestres e Segurança Operacional: Em *Layout Planning* de instalações densas (fábricas, terminais de passageiros), a ABM simula o movimento individual considerando a física de multidões e a tomada de decisão humana. Permite otimizar rotas de evacuação, dimensionar saídas de emergência e reduzir o tempo de trânsito em trocas de turno, garantindo a conformidade com normas de segurança (SSMA) [HELBING; MOLNÁR, 1995];
- Dinâmica de Mercados e Estratégia Competitiva: A ABM é empregada para criar laboratórios econômicos artificiais, onde empresas (agentes) competem por recursos e *market share*. Permite analisar a evolução de preços, a formação de cartéis ou a estabilidade de alianças estratégicas, fornecendo *insights* sobre como comportamentos microeconômicos geram padrões macroeconômicos de volatilidade [TESFATSION, 2006].

A relevância da ABM para a Engenharia de Produção reside na sua capacidade de gerenciar a complexidade. Em um cenário de personalização em massa e redes colaborativas, a ABM oferece uma lente analítica indispensável para o engenheiro projetar sistemas que não sejam apenas eficientes, mas também robustos e adaptáveis a perturbações [RAILSBACK; GRIMM, 2019].

2.3.3 Ligação com a Formação do Engenheiro de Produção

A apropriação dos princípios da Modelagem Baseada em Agentes (ABM) promove uma mudança de paradigma cognitiva no engenheiro de produção, deslocando o foco do pensamento hierárquico (*Top-Down*) para uma mentalidade descentralizada e emergente (*Bottom-Up*). O estudante aprende a reconhecer que a ordem e a eficiência em sistemas complexos — como cadeias de suprimentos globais ou o tráfego urbano — não dependem necessariamente de um controlador central onisciente, mas emergem da auto-organização de múltiplos atores autônomos. Essa perspectiva é vital para projetar sistemas resilientes, capazes de se adaptar espontaneamente a perturbações sem colapsar [BONABEAU, 2002].

A ABM também instrumentaliza o engenheiro para a análise de complexidade e não-linearidade. Ao compreender que "o todo é diferente da soma das partes", o profissional torna-se capaz de identificar riscos sistêmicos ocultos, como *tipping points* (pontos de virada) e efeitos cascata, que modelos agregados tradicionais falham em detectar. Essa competência transforma o engenheiro em um gestor de riscos robusto, apto a distinguir entre sistemas que são apenas robustos e aqueles que são antifrágeis [MACAL; NORTH, 2010].

Um diferencial estratégico desta abordagem é o desenvolvimento da competência de Design de Mecanismos e Incentivos. Como a ABM exige a modelagem explícita das regras de decisão e objetivos dos agentes, o engenheiro aprende a atuar como um "arquiteto de escolhas". Ele desenvolve a habilidade de projetar protocolos, regras de mercado e sistemas de incentivos que induzam agentes autônomos (sejam motoristas de aplicativo, fornecedores ou robôs) a cooperarem para atingir metas globais de eficiência, alinhando interesses individuais e coletivos [RAILSBACK; GRIMM, 2019].

Além disso, a ABM combate a "Falácia da Média", preparando o engenheiro para lidar com a heterogeneidade granular. Em um mercado dominado pelo *Big Data*, saber que "o cliente médio não existe" é crucial. A ABM ensina a modelar a diversidade real de comportamentos, preferências e capacidades, sendo uma ferramenta essencial para estratégias de customização em massa (*Mass Customization*) e gestão da cauda longa, onde a vantagem competitiva reside em atender às especificidades, não apenas ao padrão [KIESLING et al., 2012].

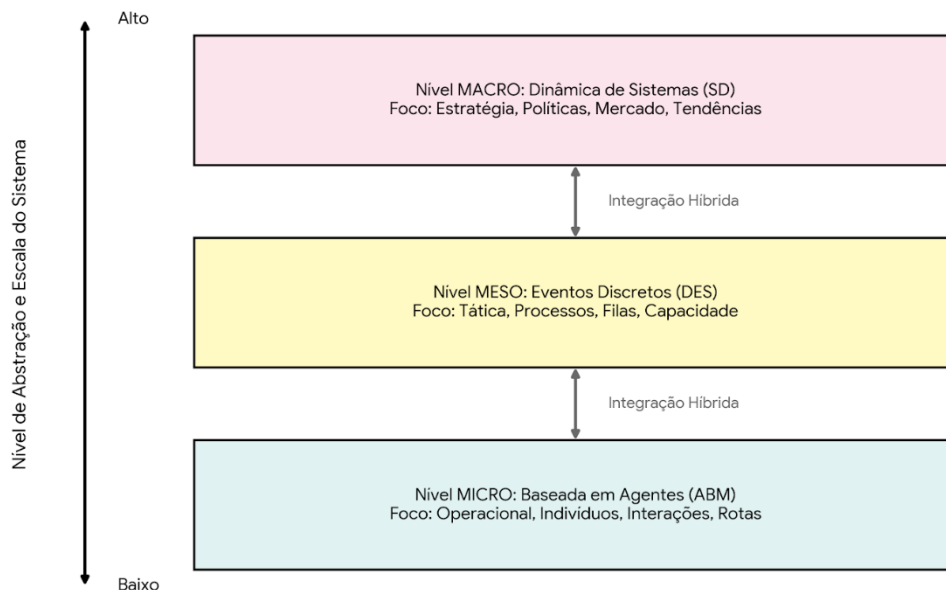
Por fim, a familiaridade com a lógica dos agentes é o alicerce para a Governança Algorítmica e a Indústria 4.0. À medida que sistemas ciber-físicos e Inteligência Artificial assumem papéis decisórios, o engenheiro de produção precisa compreender como algoritmos (agentes artificiais) interagem entre si e com humanos. A ABM oferece o laboratório ideal para testar e validar esses ambientes híbridos, capacitando o profissional a projetar, implementar e auditar ecossistemas produtivos inteligentes, descentralizados e autônomos [MONOSTORI et al., 2016].

2.4 Simulação Híbrida e Multimethod Modeling

Embora os paradigmas de Eventos Discretos (DES), Sistemas Dinâmicos (SD) e Modelagem Baseada em Agentes (ABM) tenham evoluído academicamente como disciplinas isoladas, a complexidade intrínseca dos sistemas de engenharia modernos frequentemente desafia a capacidade de representação de um único método. A Simulação Híbrida, ou *Multimethod Modeling*, refere-se à prática arquitetural de combinar diferentes paradigmas de simulação em um único modelo integrado, permitindo capturar simultaneamente múltiplos níveis de abstração e diferentes naturezas de comportamento [BORSHCHEV, 2013].

Na Engenharia de Produção, a necessidade dessa abordagem surge quando o sistema analisado possui componentes que operam em escalas distintas. A literatura clássica classifica os métodos em um espectro de abstração, sintetizado na Figura 2.4:

Figura 2.4: Espectro de Abstração na Simulação Híbrida (Multimethod)



Fonte: Elaboração do autor (2026), adaptado de Borshchev (2013).

- Nível Micro (ABM): Focado no comportamento individual e interação local (ex: comportamento de um navio específico);

- Nível Meso (DES): Focado em processos, filas e recursos (ex: operação de um terminal de carga);
- Nível Macro (SD): Focado em políticas globais e tendências agregadas (ex: economia portuária e demanda de mercado).

A modelagem híbrida permite conectar esses mundos. No contexto de um sistema logístico portuário, por exemplo, a abordagem puramente baseada em processos (DES) seria eficiente para modelar a fila e a ocupação dos berços, mas falharia em capturar a inteligência de navegação e as características únicas de cada embarcação. Por outro lado, uma abordagem puramente baseada em agentes (ABM) seria excelente para a movimentação dos navios no espaço (GIS), mas ineficiente para descrever a lógica sequencial de carga e descarga.

A solução híbrida — suportada nativamente por ambientes como o AnyLogic — permite que o engenheiro construa uma arquitetura onde Agentes (navios) fluem através de um Diagrama de Processos (DES). Dessa forma, o modelo herda a precisão processual da DES (gerenciamento de recursos, tempos de serviço) e a riqueza comportamental da ABM (atributos individuais, rotas autônomas), eliminando a necessidade de simplificações excessivas que comprometeriam a validade do estudo.

Essa sinergia metodológica é o alicerce para a criação de Gêmeos Digitais (*Digital Twins*). Ao unir a estrutura híbrida com a alimentação contínua de dados empíricos e históricos (como registros de sensores ou bases de dados governamentais), o modelo deixa de ser uma representação estática para se tornar uma réplica virtual dinâmica do sistema físico. Isso eleva substancialmente a confiabilidade da simulação como ferramenta de suporte à decisão, permitindo análises preditivas e prescritivas com um grau de fidelidade inalcançável por métodos isolados.

3 O AMBIENTE PARA AS SIMULAÇÕES (MATERIAIS E MÉTODOS)

O presente capítulo estabelece a conexão entre os fundamentos teóricos apresentados no Capítulo 2 e sua materialização em um ambiente computacional de simulação. O objetivo não é fornecer um tutorial operacional, mas sim apresentar uma

análise arquitetural do software *AnyLogic*, explicitando como seus componentes estruturais implementam os paradigmas de Simulação de Eventos Discretos (DES), Sistemas Dinâmicos (SD) e Modelagem Baseada em Agentes (ABM).

Compreender a lógica subjacente à ferramenta é essencial para que o estudante de engenharia seja capaz de selecionar, de forma fundamentada, os módulos adequados para cada classe de problema, transcendendo o simples domínio operacional e desenvolvendo competência analítica na construção de modelos.

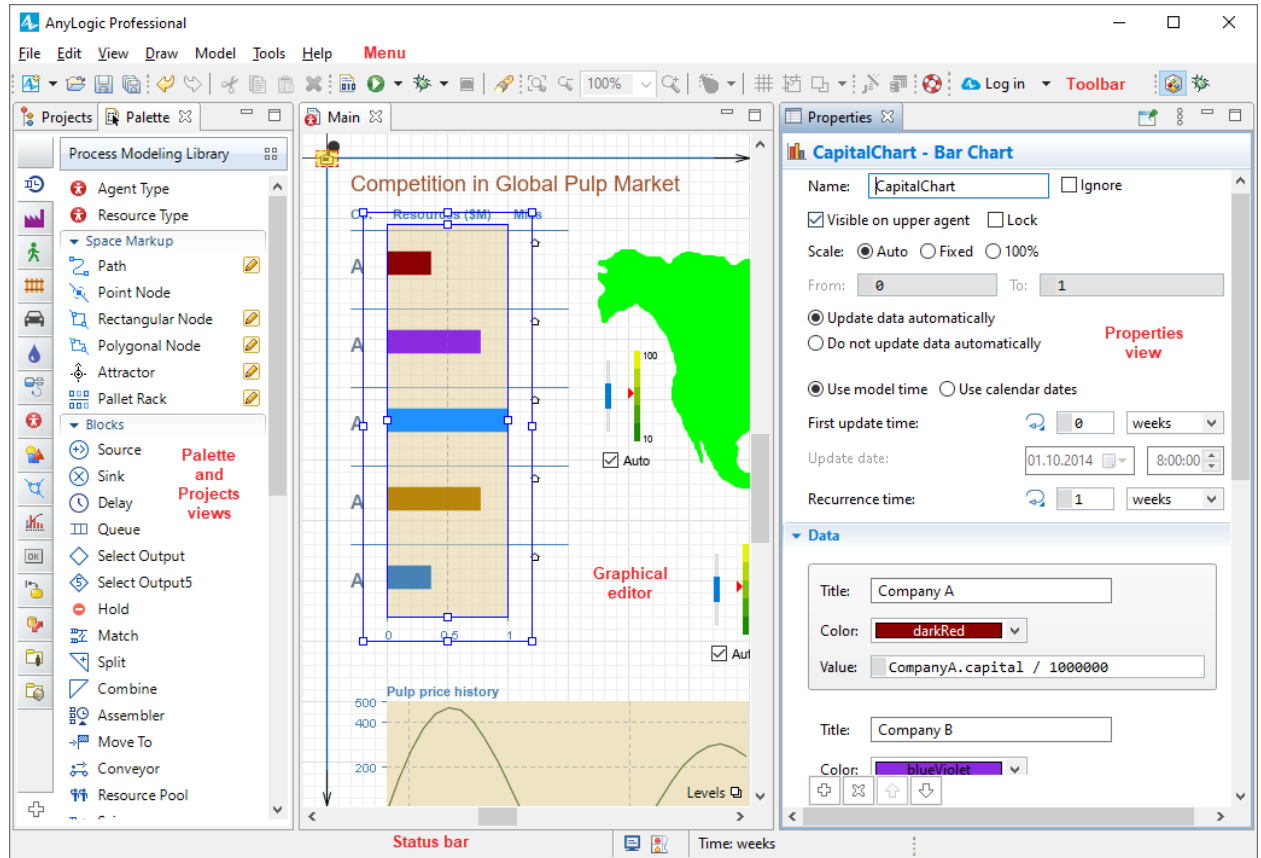
3.1 A Interface do *AnyLogic* e a Fundamentação Orientada a Objetos

O *AnyLogic* adota um paradigma de desenvolvimento visual, no qual modelos de simulação são construídos através da composição gráfica de elementos diagramáticos, ao invés da codificação textual de algoritmos. Essa abordagem alinha-se com a natureza inerentemente visual dos três paradigmas de modelagem: fluxogramas de processos (DES), diagramas de estoque e fluxo (SD) e redes de agentes interativos (ABM).

A interface foi projetada para facilitar a tradução direta de conceitos teóricos em representações computacionais executáveis. [GRIGORYEV,2018]. A área de trabalho estrutura-se em quatro regiões funcionais fixas, que organizam o processo de modelagem:

- *Projects View*: painel hierárquico que apresenta a estrutura lógica do modelo, incluindo agentes, experimentos, classes auxiliares e recursos gráficos;
- *Palette View*: biblioteca visual de elementos de modelagem, organizados por paradigma e funcionalidade. Contém as *palettes* específicas para *Process Modeling* (DES), *System Dynamics* (SD), *Agent* (ABM), além de bibliotecas para análise estatística e visualização 3D;
- *Graphical Editor*: área central de trabalho onde os diagramas são efetivamente construídos. Suporta múltiplas camadas de visualização (lógica de processo, *layout* espacial, animação) e permite a alternância entre diferentes níveis de abstração;
- *Properties View*: painel contextual que exhibe e permite a edição dos atributos do elemento selecionado. Aqui são definidos parâmetros quantitativos (taxas, capacidades, distribuições de probabilidade) e conectividades lógicas.

Figura 3.1: Interface de desenvolvimento do AnyLogic, destacando as áreas de (1) Projetos, (2) Paleta de Bibliotecas, (3) Editor Gráfico e (4) Propriedades. *Fonte: The AnyLogic Company.*



Fonte: The AnyLogic Company (2026).

Além da interface visual, um diferencial arquitetural primário do *AnyLogic* é sua fundamentação nativa na Linguagem Java e na Orientação a Objetos. Cada agente modelado é, tecnicamente, uma Classe Java. Isso permite o uso de conceitos avançados de engenharia de software, como herança (criar um agente "Veículo" genérico e subagentes "Caminhão" e "Navio" que herdem suas propriedades) e polimorfismo. Isso garante extensibilidade total: onde os blocos gráficos atingem seu limite, o engenheiro pode inserir código Java nativo para modelar lógicas de decisão complexas ou conectar-se a bibliotecas externas [GRIGORYEV, 2018, p. 28].

Quanto aos modos de execução, o *AnyLogic* diferencia dois regimes temporais:

- *Virtual Time*: modo padrão para experimentação analítica, no qual o relógio de simulação avança na velocidade máxima de processamento da CPU. Adequado para cenários de longo prazo e análises de sensibilidade.
- *Real Time*: modo no qual o tempo de simulação é sincronizado com o tempo real, útil para validação visual (*face validity*) e apresentações interativas.

Para garantir a confiabilidade técnica (V&V), a interface inclui ferramentas de Depuração em Tempo de Execução. O engenheiro pode inspecionar o estado de qualquer agente "ao vivo", visualizar logs de eventos passo a passo e utilizar o console Java para rastrear erros lógicos antes da execução dos experimentos finais.

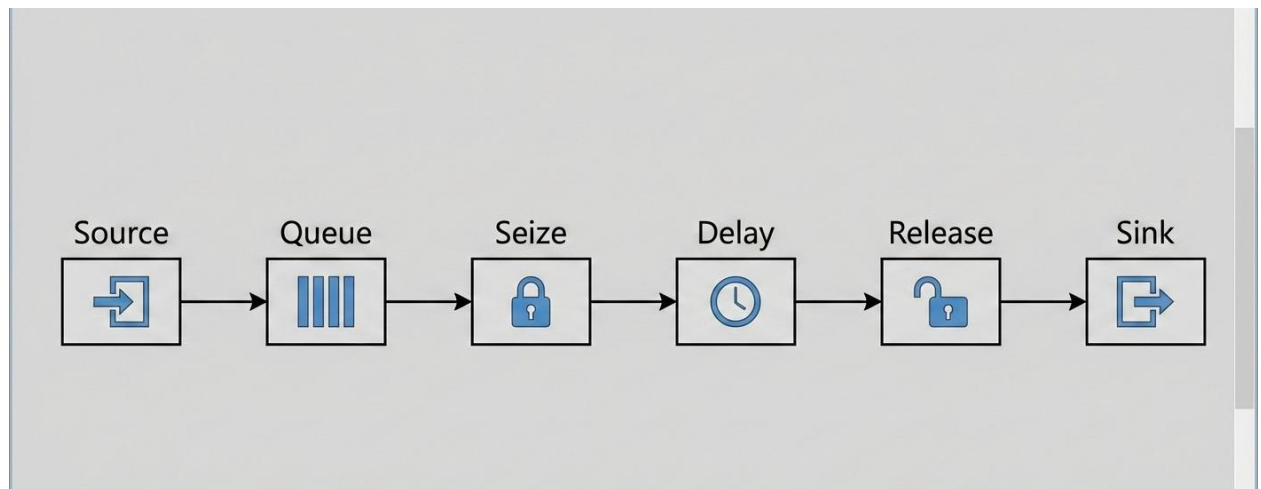
3.2 Módulos para a Simulação de Eventos Discretos

A *Process Modeling Library* constitui o núcleo funcional para a implementação de modelos DES no *AnyLogic*. Essa biblioteca traduz os conceitos estruturais da simulação discreta em componentes gráficos interconectáveis. [GRIGORYEV, 2018, p. 134]. A correspondência entre os elementos teóricos e sua materialização no software é sistematizada a seguir:

- Geração e destruição de entidades: os blocos *Source* e *Sink* controlam a criação e remoção de entidades. O bloco *Source* permite a especificação de distribuições de probabilidade para intervalos entre chegadas (exponencial, normal, empírica), viabilizando a modelagem da variabilidade estocástica;
- Gestão de recursos: a tripla de blocos *ResourcePool*, *Seize* e *Release* implementa o paradigma de alocação de recursos. O *ResourcePool* define unidades com capacidade finita (máquinas, operadores). As entidades solicitam a alocação através do *Seize* (formando fila se ocupado) e liberam através do *Release*;
- Filas e atrasos: o bloco *Queue* representa áreas de acumulação com disciplina configurável (*FIFO*, *LIFO*, prioridade). O bloco *Delay* impõe um atraso temporal, modelando tempos de processamento ou transporte;
- Lógica de Decisão e Roteamento: o bloco *SelectOutput* implementa a ramificação de fluxo. Permite que entidades sigam caminhos distintos baseados em probabilidades (estocástico) ou em condições booleanas determinísticas (ex: agente.falha == true), fundamental para triagem e controle de qualidade;

- Agrupamento (Lotes): os blocos *Batch* e *Unbatch* modelam a transformação de múltiplas entidades em uma única (e vice-versa), permitindo a representação fiel de processos de paletização e consolidação de carga em terminais;
- Movimentação e armazenamento: blocos como *MoveTo*, *RackStore* e *RackPick* introduzem a dimensão espacial, integrando a lógica do processo à geografia do *layout* físico.

Figura 3.2: Exemplo de modelagem de um processo de atendimento simples utilizando os blocos da *Process Modeling Library* do AnyLogic. As entidades chegam (*Source*), aguardam em fila (*Queue*), capturam um recurso (*Seize*), são processadas (*Delay*), liberam o recurso (*Release*) e saem do sistema (*Sink*)



Fonte: Elaboração do autor (2026).

Um diferencial arquitetural é a integração nativa com o espaço 2D/3D. Os elementos da *Space Markup* (caminhos, nós, atratores) permitem definir a topologia física sobre a qual as entidades se movem. Essa geografia é automaticamente considerada nos cálculos de tempo de transporte, enriquecendo a fidelidade do modelo.

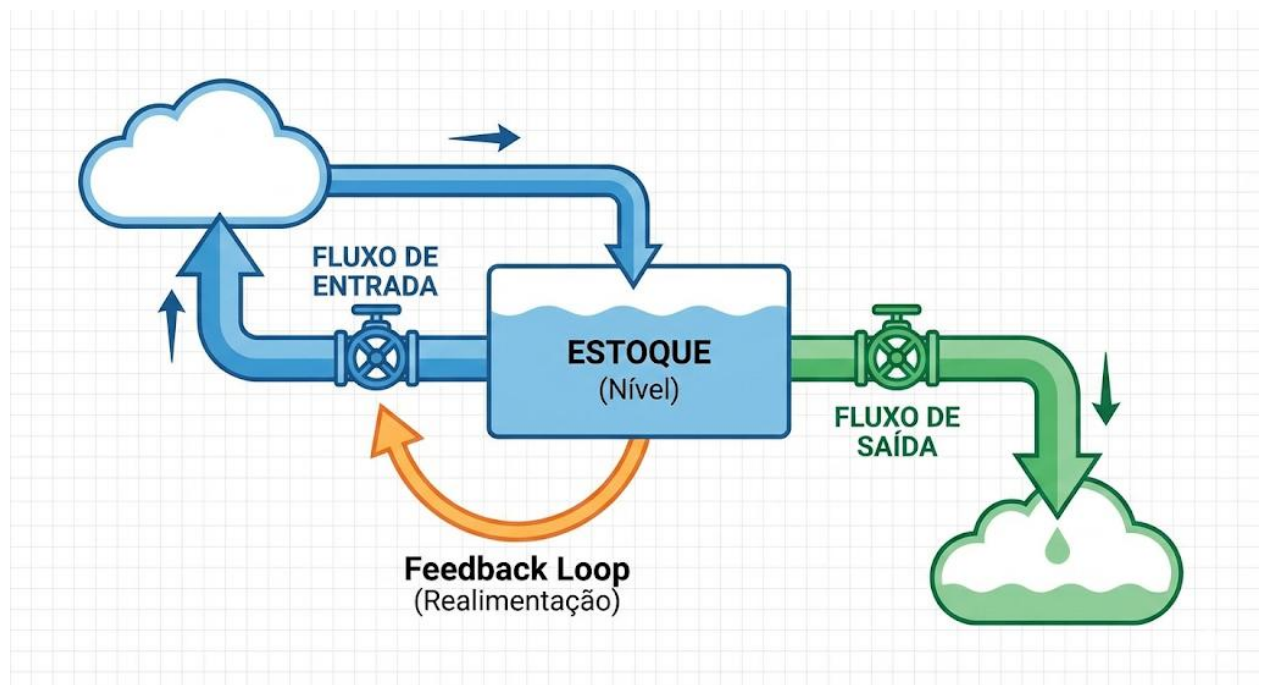
3.3 Módulos para a Simulação de Sistemas Dinâmicos

A *System Dynamics Palette* permite traduzir diagramas causais e de estoque-fluxo diretamente em modelos executáveis, abstraindo a manipulação explícita de equações diferenciais apresentados na Seção 2.2. [GRIGORYEV, 2018, p. 103].

Os elementos fundamentais são:

- *Stock* (Estoque): representa uma variável de acumulação (integral no tempo). Armazena quantidades como inventário ou capital;
- *Flow* (Fluxo): representa uma taxa de variação (derivada). É governado por uma equação algébrica e altera o nível dos estoques conectados;
- *Variable* e *Parameter*: elementos auxiliares que processam informações e definem constantes exógenas, explicitando as regras de decisão;
- *Link* e *Loop Indicator*: conectores que representam dependências causais e identificadores visuais de laços de realimentação (Reforço/Equilíbrio).

Figura 3.3: Representação conceitual do SD. O diagrama ilustra a física fundamental do método, onde Estoques (retângulos) acumulam o resultado líquido de Fluxos de entrada e saída (válvulas), governados por laços de realimentação (*feedback loops*) que buscam o equilíbrio do sistema.



Fonte: Elaboração do autor (2026).

Uma distinção arquitetural fundamental da implementação de *SD* no *AnyLogic* reside no tratamento implícito das equações diferenciais. Diferente da modelagem algébrica tradicional em planilhas eletrônicas (como o *Excel*), onde o analista é forçado a discretizar manualmente o tempo — transformando integrais em somas cumulativas com passo fixo (Δt) —, o ambiente do *AnyLogic* preserva a semântica contínua das variáveis de fluxo [STERMAN, 2000].

O motor de simulação gerencia autonomamente a integração numérica, aplicando algoritmos adaptativos (como solucionadores baseados em *Runge-Kutta*) que ajustam dinamicamente o passo de tempo para garantir a precisão matemática sem intervenção do usuário. Essa abstração técnica elimina a necessidade de configuração manual de métodos numéricos (como o método de *Euler*), permitindo que o engenheiro direcione seu esforço cognitivo para a validação das hipóteses causais e a estrutura lógica do modelo, em vez de se deter na complexidade da implementação computacional [GRIGORYEV, 2018, p. 110].

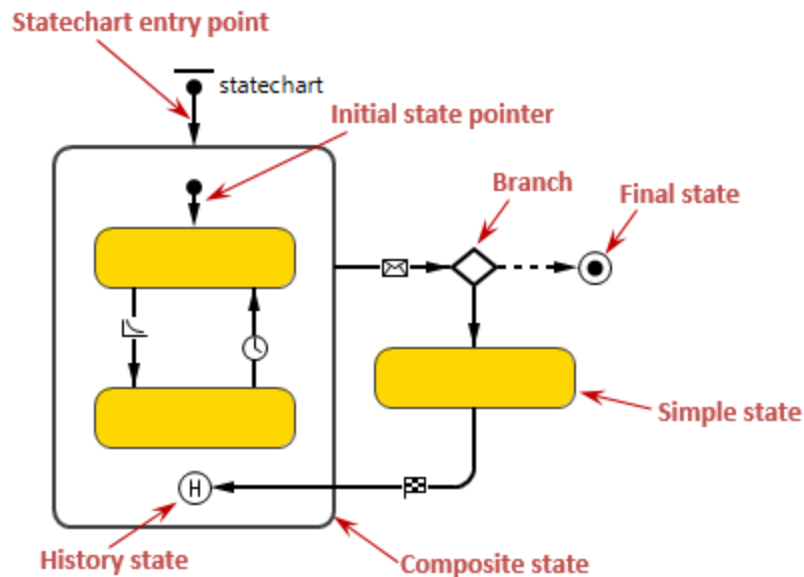
3.4 Módulos para a Modelagem Baseada em Agentes

A ABM adota uma abordagem *bottom-up*, onde o comportamento global emerge das interações locais entre agentes autônomos. O *AnyLogic* estrutura essa construção através de:

- **Parâmetros e Variáveis:** definem a heterogeneidade da população (cada agente tem seus próprios atributos, como velocidade ou capacidade de carga);
- **Statecharts (Máquinas de Estado):** diagrama central que implementa a lógica comportamental e reativa. Estados discretos (ex: "Atracado", "Navegando", "Em Manutenção") são conectados por transições disparadas por eventos, mensagens ou tempo;
- **Comunicação:** mecanismos de mensagens assíncronas permitem que agentes negociem e troquem informações, viabilizando a modelagem de sistemas descentralizados;
- **Ambiente e GIS:** o *AnyLogic* oferece suporte a espaços geográficos referenciados (*GIS*). Diferente de um plano cartesiano simples, o ambiente GIS utiliza servidores

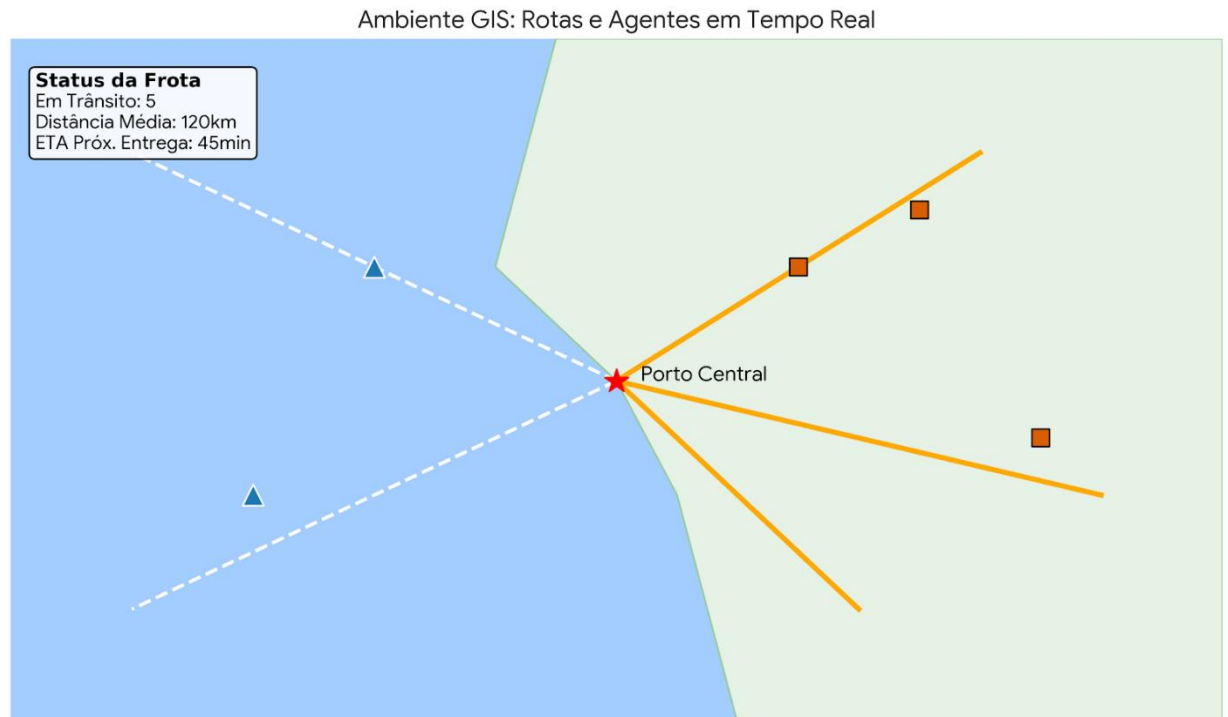
de mapas (como *OpenStreetMap*) para calcular rotas reais em malhas rodoviárias, ferroviárias ou marítimas. Para a engenharia logística, isso eleva substancialmente a precisão do cálculo de distâncias e tempos de trânsito.

Figura 3.4: Exemplo de um diagrama de Máquina de Estados (*Statechart*) no AnyLogic, utilizado para modelar o comportamento reativo de um agente. Os retângulos representam os estados possíveis e as setas indicam as transições disparadas por eventos ou condições. *Fonte: The AnyLogic Company.*



Fonte: The AnyLogic Company (2026).

Figura 3.5: Ambiente de simulação integrado a Sistemas de Informação Geográfica (GIS). Diferente de um plano cartesiano abstrato, os agentes (navios e caminhões) navegam por rotas reais georreferenciadas, permitindo o cálculo preciso de distâncias e tempos de trânsito baseados na topologia da malha logística existente. *Fonte: Elaboração do autor (2026).*



Fonte: O autor, gerado no software AnyLogic PLE.

3.5 Simulação Híbrida (*Multimethod Modeling*)

Sistemas reais de engenharia frequentemente exigem a integração de múltiplos paradigmas. O *AnyLogic* viabiliza a simulação híbrida através de três mecanismos fundamentais:

1. Agentes como contêineres multimétodo: um único agente pode conter lógicas de DES, SD e ABM simultaneamente. Por exemplo, um "Agente Porto" pode ter um fluxograma discreto para o carregamento de navios e um diagrama de SD para o controle financeiro;

2. Acoplamento bidirecional: variáveis de um paradigma alimentam parâmetros de outro (ex: o tamanho da fila no DES influencia a taxa de investimento no SD);
3. Sincronização temporal unificada: o motor gerencia uma fila global de eventos que ordena cronologicamente as ações discretas e os passos de integração contínua.

Essa capacidade é vital para evitar conclusões equivocadas decorrentes de análises isoladas em uma única escala (operacional, tática ou estratégica).

3.6 Dados, Conectividade e Experimentos de Otimização

A utilidade de um modelo na Engenharia de Produção depende de sua alimentação por dados reais e da capacidade de otimização. O ambiente integra-se ao ecossistema de dados através de:

- *Input Data* (Conectividade): leitura nativa de arquivos *Excel*, *CSV* e conexão com bancos de dados *SQL/JDBC*. Isso permite que o modelo seja "data-driven" (guiado por dados), onde a estrutura da simulação é parametrizada dinamicamente pela base de dados carregada na inicialização;
- *Output Analysis*: exportação de logs e estatísticas para ferramentas de BI (como *Power BI* ou *Tableau*) e geração de *dashboards* internos.

Além da execução padrão, a arquitetura suporta Experimentos Avançados:

1. Análise de Sensibilidade (Monte Carlo): execução automática de múltiplas replicações com sementes aleatórias distintas, permitindo a construção de intervalos de confiança estatísticos para validar a robustez do sistema frente à incerteza;
2. Otimização (*OptQuest*): integração nativa com o motor heurístico *OptQuest*. O algoritmo busca automaticamente a combinação ótima de parâmetros de entrada (ex: número ideal de portêineres) que maximiza uma função-objetivo (ex: lucro ou vazão), respeitando as restrições operacionais.

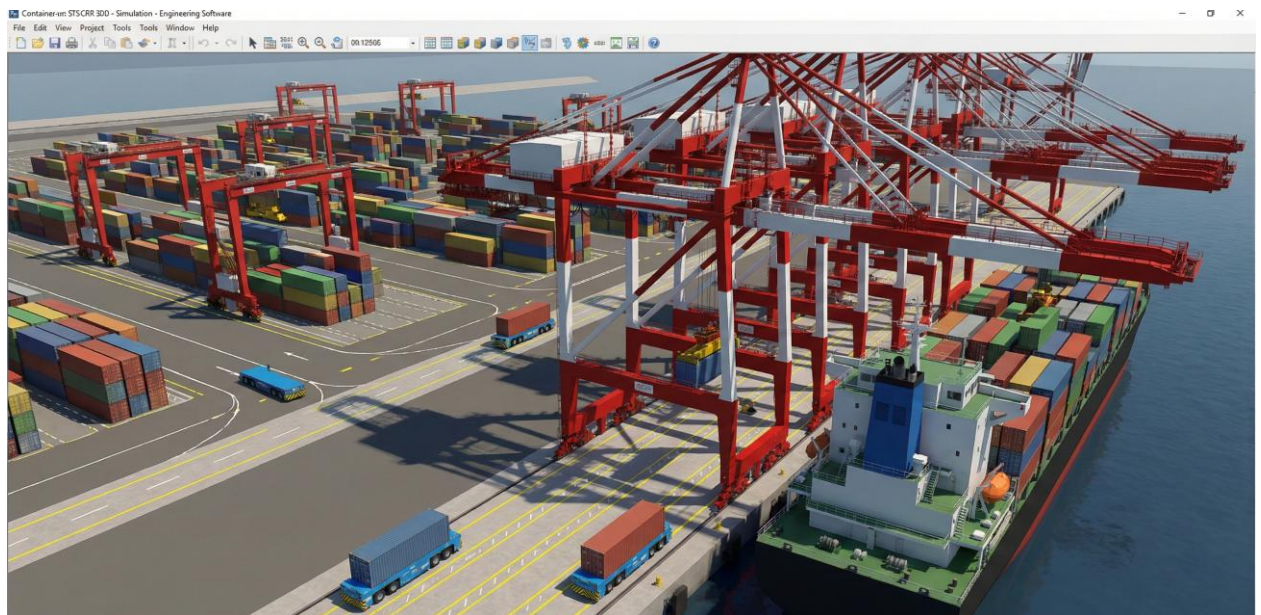
3.7 Bibliotecas Específicas de Domínio (*Material Handling* e *Fluid*)

Para elevar a fidelidade física em projetos de infraestrutura, o *AnyLogic* oferece bibliotecas verticais especializadas, essenciais para o contexto portuário:

- *Material Handling Library*: simula a física detalhada de movimentação discreta. Inclui blocos para Esteiras Transportadoras (*Conveyors*), Pontes Rolantes

(Cranes) e AGVs, considerando aceleração, colisão e dimensões reais da carga. Fundamental para terminais de contêineres;

Figura 3.6: Visualização 3D de alta fidelidade de um terminal de contêineres simulado. A utilização da *Material Handling Library* permite representar a física real de equipamentos como portêineres (STS) e veículos guiados (AGVs), facilitando a validação visual do modelo (*Face Validity*) e a análise de gargalos espaciais no pátio. Fonte: Elaboração do autor (2026).



Fonte: O autor, gerado no software AnyLogic PLE.

- *Fluid Library*: simula o fluxo de materiais contínuos e a granel. Permite modelar Tanques, Tubulações e Válvulas com taxas de fluxo variáveis. Indispensável para a simulação de terminais de granéis líquidos e sólidos, integrando a dinâmica de fluidos com a logística discreta de chegada dos navios;
- *Rail Library*: modelagem microscópica de tráfego ferroviário, considerando o acoplamento de vagões e a topologia dos trilhos no pátio logístico.

3.8 Entrega via *Cloud Computing*

Em consonância com a Indústria 4.0, o software suporta a exportação de modelos para o *AnyLogic Cloud*. Esta plataforma web permite que *stakeholders* e gestores executem cenários e visualizem resultados diretamente no navegador, sem necessidade de instalação local. Isso democratiza o acesso à ferramenta de apoio à decisão e facilita a colaboração remota.

Outro diferencial técnico e gerencial do software é a sua integração nativa com o *AnyLogic Cloud*. Essa plataforma baseada em nuvem permite que os modelos desenvolvidos e validados no ambiente *desktop* do engenheiro sejam exportados e hospedados em servidores web. Do ponto de vista prático e de negócios, isso significa que tomadores de decisão, clientes finais ou diretores não necessitam instalar o software de simulação em suas máquinas, tampouco possuir conhecimentos em programação Java ou modelagem. Através de um link seguro, o gestor acessa o modelo diretamente pelo navegador, interage com *dashboards* personalizados, altera parâmetros (como o aumento de demanda ou o tamanho da frota de caminhões) e executa experimentos de otimização utilizando o alto poder de processamento da nuvem. Essa capacidade de entrega (*deployment*) transforma o modelo computacional de um artefato estritamente técnico em uma ferramenta executiva dinâmica para a tomada de decisão.

3.9 Integração com Inteligência Artificial e *Python*

Reconhecendo a convergência entre Simulação e Ciência de Dados, a arquitetura recente do *AnyLogic* permite a interoperabilidade com a linguagem *Python*. Através do conector *Pypeline* (ou integrações nativas), o modelo de simulação pode atuar como um "ambiente de treinamento" digital para algoritmos de Aprendizado por Reforço (*Reinforcement Learning*).

Nessa configuração, o *AnyLogic* fornece o estado do sistema e a recompensa simulada, enquanto um agente externo de IA (treinado em bibliotecas como *TensorFlow* ou *PyTorch*) toma as decisões de controle otimizadas. Essa capacidade posiciona a ferramenta como uma plataforma essencial para o desenvolvimento e validação segura de IAs industriais antes de sua implantação física.

4 ESTUDO DE CASO: MODELAGEM ESTOCÁSTICA DE UMA REDE LOGÍSTICA REGIONAL COM ROTEAMENTO GIS E CONTROLE KANBAN

4.1 Considerações Iniciais e Objetivos do Estudo

A consolidação da aprendizagem em simulação computacional exige a transposição dos conceitos teóricos para problemas que capturem a complexidade e a interdependência dos sistemas reais. Neste capítulo, apresenta-se o desenvolvimento de um modelo de simulação híbrido focado em Gestão da Cadeia de Suprimentos (*Supply Chain Management*), desenvolvido na versão educacional do software *AnyLogic* (*Personal Learning Edition*).

O objeto de estudo é uma rede logística multinível situada na Europa Central, abrangendo operações de manufatura, armazenagem e distribuição capilar entre a Holanda e a Alemanha. A escolha deste cenário é intencional e pedagógica, visando demonstrar a capacidade da ferramenta em unificar três dimensões críticas para a formação do Engenheiro de Produção:

1. Dimensão Espacial (GIS): A influência das distâncias reais e da topologia rodoviária nos tempos de ciclo logístico (*Lead Time*).
2. Dimensão Operacional (DES): O gerenciamento de recursos finitos (capacidade de frota) e a variabilidade estocástica dos processos.
3. Dimensão Tática (Agentes): A coordenação descentralizada do fluxo de materiais através de uma lógica "puxada" (*Pull System*), alinhando a produção à demanda real.

4.2 Caracterização do Sistema e Parâmetros de Entrada

O modelo representa uma cadeia de abastecimento composta por três elos hierárquicos, estruturados computacionalmente como agentes autônomos. Para garantir a viabilidade didática do estudo, foram adotados parâmetros operacionais hipotéticos, porém verossímeis, descritos a seguir:

4.2.1 Estrutura dos Agentes

- Elo *Upstream* (Produção): Representado pelo agente *Producer*, localizado na cidade de Groningen. O sistema opera sob lógica *Make-to-Order* (Produção sob Pedido), com tempo de *setup* e produção modelado por uma distribuição Exponencial (média de μ horas), refletindo a variabilidade natural de chão de fábrica.
- Elo Intermediário (Distribuição): O agente *Wholesaler* (Atacadista), situado estrategicamente no porto de Eemshaven, atua como *Hub* de consolidação e desconsolidação de carga (*Break-bulk*).
- Elo *Downstream* (Varejo): Quatro agentes *Retailers* (Varejistas) distribuídos em Drachten, Assen, Leeuwarden (Holanda) e Leer (Alemanha), representando os pontos de consumo.

4.2.2 Parâmetros da Frota e Demanda

- Geração de Pedidos: A demanda é a fonte primária de aleatoriedade do sistema. Cada varejista gera pedidos seguindo uma taxa de chegada (*Rate*) de 7 pedidos/dia, introduzindo incerteza no planejamento da distribuição.
- Tempos Operacionais: As operações de *Loading* (carregamento) e *Unloading* (descarregamento) nos terminais seguem uma distribuição Uniforme (entre 2 e 3 horas), representando a variação na eficiência das equipes operacionais.
- Velocidade da Frota: Os caminhões operam com velocidade máxima de 80 km/h, sujeita às restrições reais de velocidade das vias importadas do mapa GIS.

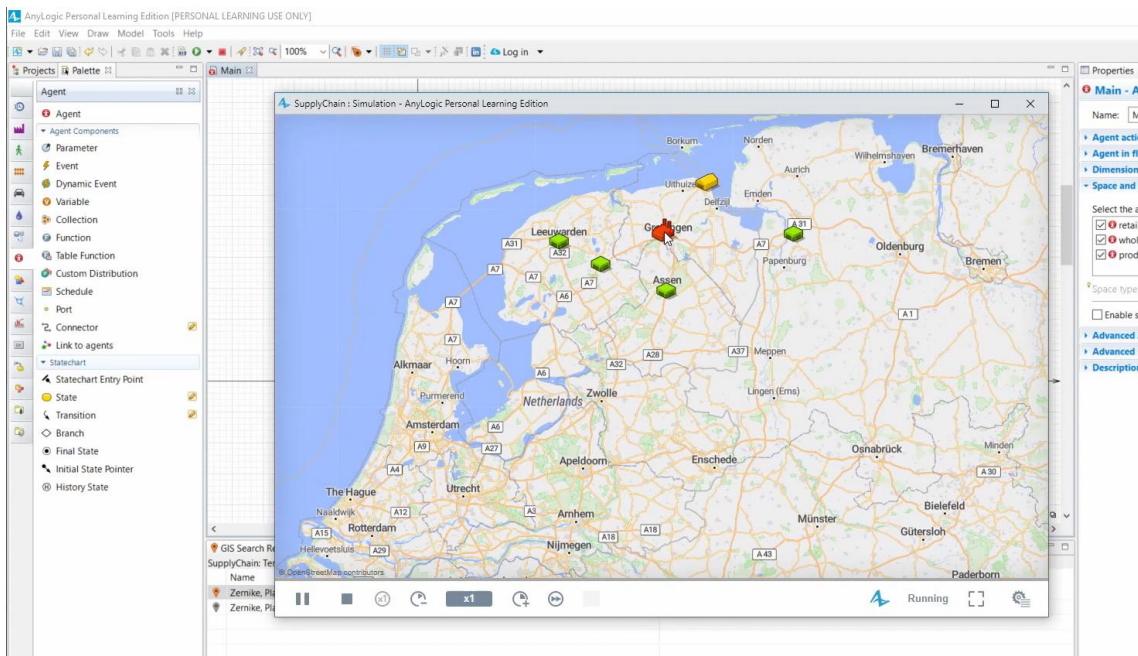
4.3 Metodologia de Implementação Híbrida

A construção do modelo seguiu uma abordagem incremental, integrando os paradigmas de ABM, DES e GIS conforme a metodologia proposta no Capítulo 3.

4.3.1 Roteamento em Malha Viária Real (GIS)

Diferente de abordagens tradicionais que simplificam o transporte assumindo distâncias euclidianas (linha reta), este estudo utilizou o motor de roteamento nativo do *AnyLogic*. Ao conectar-se aos servidores do *OpenStreetMap*, a modelo baixa a topologia real das rodovias. O algoritmo calcula o caminho mais rápido considerando-nos e arestas da malha viária, conferindo alta fidelidade ao cálculo dos tempos de deslocamento, especialmente para rotas transfronteiriças.

Figura 4.1: Configuração espacial da rede logística no ambiente GIS, evidenciando a dispersão geográfica entre os elos da cadeia. Fonte: O autor, gerado no software *AnyLogic PLE*.



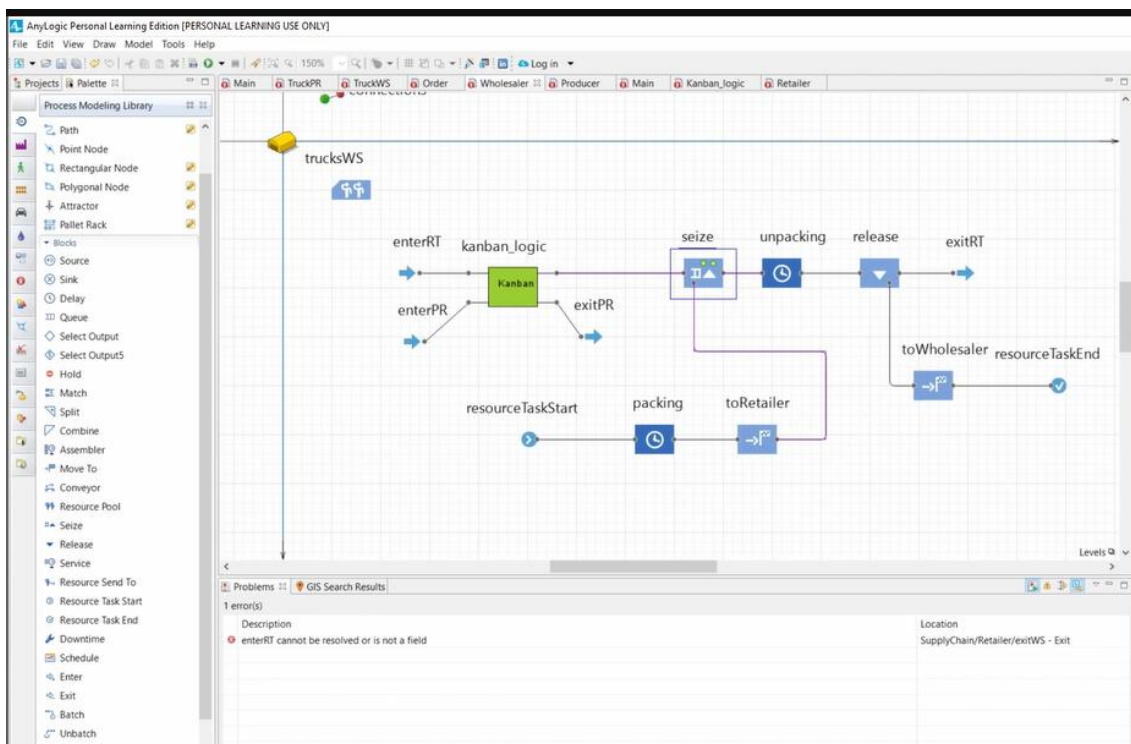
Fonte: O autor, gerado no software *AnyLogic PLE*.

4.3.2 Modelagem de Processos Internos (DES)

A lógica operacional foi implementada através da *Process Modeling Library*. No agente Atacadista, um fluxograma de Eventos Discretos gerencia o ciclo do pedido:

1. Recepção (Enter): Entrada da ordem digital enviada pelo varejo.
2. Captura de Recurso (Seize): Tentativa de alocação de um caminhão da frota TruckFleet. A capacidade finita da frota gera filas de espera (*Backlog*) em momentos de pico de demanda.
3. Movimentação (MoveTo): Acionamento do motor GIS para deslocamento físico do agente até o destino.

Figura 4.2: Diagrama de processos internos do Atacadista, detalhando a lógica sequencial de alocação de frota e expedição. Fonte: O autor, gerado no software AnyLogic PLE.



Fonte: O autor, gerado no software AnyLogic PLE.

4.3.3 Lógica de Controle: Kanban Eletrônico

Para mitigar o Efeito Chicote (*Bullwhip Effect*), implementou-se uma política de estoque baseada em Kanban. O fluxo de informação ocorre no sentido inverso ao fluxo de

materiais: a saída de um lote do Atacadista gera um sinal instantâneo para o Produtor, que somente então inicia a reposição. Essa lógica garante que o sistema opere com níveis de estoque otimizados, eliminando a superprodução característica de sistemas "empurrados" (*Push*).

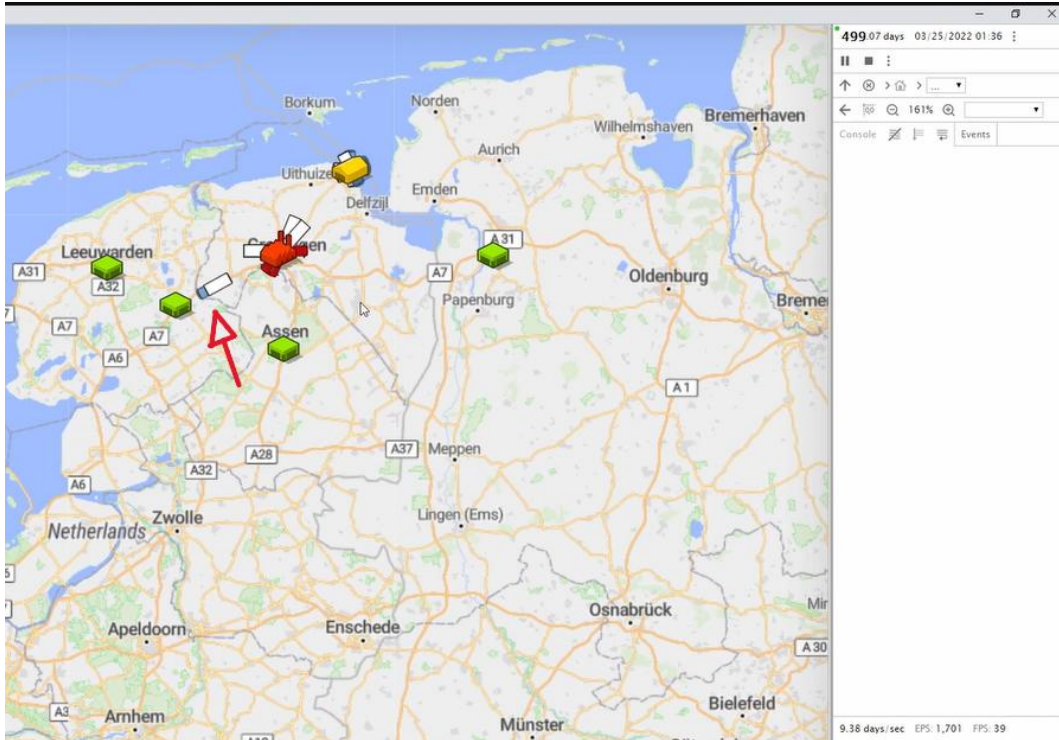
4.4 Verificação, Validação e Análise de Resultados

O modelo foi submetido a técnicas de Verificação (depuração do código e rastreamento de entidades) e Validação de Face (confirmação visual do comportamento dos agentes durante a execução) antes da coleta de dados.

4.4.1 Análise do Roteamento Dinâmico

A simulação em modo *Real Time* revelou o impacto da geografia no desempenho logístico. Observou-se que as entregas para o varejista em Leer (Alemanha) apresentaram tempos de ciclo significativamente superiores às rotas domésticas holandesas. Em um modelo estático (planilha), essa variabilidade seria tipicamente mascarada por médias, subestimando a necessidade de frota. A simulação demonstrou que a distância geográfica afeta diretamente a Taxa de Giro dos veículos e, conseqüentemente, a capacidade de atendimento do sistema.

Figura 4.3: Execução da simulação, demonstrando o fluxo de transporte em tempo real e a aderência da frota à malha rodoviária existente. Fonte: O autor, gerado no software AnyLogic PLE.



Fonte: O autor, gerado no software AnyLogic PLE

4.4.2 Desempenho da Política Kanban e Gargalos

A análise do comportamento do sistema indicou que a política Kanban foi eficaz na sincronização entre produção e demanda, mantendo o estoque em processo (WIP) estável.

Entretanto, a simulação permitiu identificar o deslocamento do Gargalo do Sistema. Em cenários de alta demanda simultânea, a taxa de ocupação da frota do Atacadista atingiu a saturação (100%), gerando filas de pedidos aguardando transporte. Isso evidencia que, no cenário simulado, a restrição principal não é a capacidade produtiva da fábrica, mas a capacidade logística de distribuição.

4.5 Discussão e Contribuições Didáticas

4.5.1 Limitações do Modelo

Como toda representação computacional é uma abstração da realidade, cabe destacar as limitações deste estudo. No aspecto operacional, variáveis como turnos de trabalho dos motoristas, manutenções corretivas da frota e congestionamentos em tempo real não foram incorporadas, podendo ser objeto de expansão em trabalhos futuros.

No aspecto computacional, ressalta-se que o modelo foi desenvolvido utilizando a versão *Personal Learning Edition* (PLE) do software AnyLogic. A escolha dessa versão justifica-se pelo seu caráter educacional e gratuito, garantindo que estudantes possam reproduzir o estudo sem custos. Entretanto, a versão PLE possui limitações técnicas exatas (hard-coded) inerentes à sua licença. Entre as principais restrições, destacam-se: o limite estrutural de no máximo 10 tipos de agentes diferentes por modelo; um teto de 50.000 agentes gerados dinamicamente durante a execução; limite de 5 horas virtuais de simulação para bibliotecas específicas; limitações no motor de otimização *OptQuest* (restrito a 500 iterações e 7 parâmetros de decisão); e restrições na taxa de requisições de roteamento aos servidores públicos de mapas (GIS).

Para o escopo didático deste trabalho, que foca na compreensão empírica e na superação da fragmentação curricular, essas limitações de escala não inviabilizam os resultados. Contudo, evidenciam que, caso o modelo seja escalado para aplicações de nível industrial — lidando com *Big Data* ou centenas de milhares de rotas simultâneas —, seria indispensável a migração para a licença *Professional* da ferramenta.

4.5.2 Valor Didático do Artefato

Este estudo de caso cumpre o objetivo central do trabalho ao servir como um artefato didático integrador. Para o estudante de Engenharia, o modelo permite:

1. Visualizar a Interdependência: Compreender empiricamente como decisões locais (localização de um cliente) afetam métricas globais (ocupação da frota).
2. Experimentar em Ambiente Seguro: Testar o comportamento da cadeia de suprimentos sob condições de estresse (ex: picos de demanda) sem riscos financeiros.

3. Superar a Fragmentação Curricular: Unificar disciplinas tipicamente isoladas — Logística, Pesquisa Operacional e PCP — em uma única plataforma analítica visual, promovendo o pensamento sistêmico.

5 CONCLUSÃO

5.1 Síntese e Cumprimento dos Objetivos

Este trabalho originou-se da necessidade crítica de preencher uma lacuna na bibliografia técnica em língua portuguesa: a escassez de materiais didáticos estruturados para o ensino de simulação multimétodo. Ao longo desta pesquisa, buscou-se não apenas apresentar uma ferramenta de software, mas instrumentalizar o estudante de Engenharia de Produção com uma nova competência analítica.

Retomando os objetivos específicos estabelecidos no Capítulo 1, conclui-se que foram plenamente alcançados:

1. Fundamentação Teórica (Atendido no Capítulo 2): Os paradigmas de DES, SD e ABM foram desmistificados e integrados, demonstrando que a complexidade dos sistemas modernos exige uma abordagem híbrida.
2. Domínio do Ambiente (Atendido no Capítulo 3): A análise da arquitetura do *AnyLogic* provou que a plataforma oferece a flexibilidade necessária — via interface visual e extensibilidade Java — para superar as limitações dos simuladores tradicionais.
3. Aplicação Prática (Atendido no Capítulo 4): O desenvolvimento do estudo de caso da Rede Logística Regional validou a metodologia proposta. O modelo serviu como prova de conceito de que é possível unir a precisão operacional (filas e recursos) com a visão estratégica (mapas GIS e políticas Kanban) em um único ambiente didático.

Dessa forma, o trabalho entrega à comunidade acadêmica um artefato educacional robusto, capaz de reduzir a curva de aprendizado e fomentar a difusão da simulação avançada no Brasil.

5.2 Limitações do Estudo

A integridade acadêmica exige o reconhecimento das fronteiras desta pesquisa. As principais limitações identificadas foram:

- **Abstração de Custos:** O estudo de caso concentrou-se em métricas de desempenho operacional (nível de serviço, ocupação, tempo). Variáveis financeiras (CAPEX/OPEX) e custos logísticos não foram modelados, o que seria essencial para uma análise de viabilidade econômica completa.
- **Dados Determinísticos:** Para fins didáticos, utilizou-se distribuições estatísticas teóricas para a demanda e tempos de processo. Em uma aplicação industrial real, seria mandatória a etapa de *Input Modeling* baseada na mineração de dados históricos da empresa.

5.3 Recomendações para Trabalhos Futuros

A conclusão deste TCC não encerra o tema, mas estabelece uma base sólida para novas investigações. Recomenda-se:

1. **Validação Pedagógica:** Aplicar o material didático desenvolvido neste TCC em uma disciplina eletiva de graduação, coletando *feedback* dos alunos para mensurar a eficácia do aprendizado e a usabilidade dos tutoriais.
2. **Integração com Inteligência Artificial:** Explorar o conector *Pypeline* do *AnyLogic* para desenvolver modelos onde a política de controle (ex: despacho de caminhões) não seja fixa, mas aprendida dinamicamente por um agente de *Reinforcement Learning* (Python), alinhando o ensino às tendências da Indústria 4.0.
3. **Expansão Setorial:** Adaptar a metodologia para outros setores carentes de simulação no Brasil, como a modelagem de evacuação de multidões (*Pedestrian Library*) ou a otimização de pátios ferroviários (*Rail Library*).

5.4 Considerações Finais

Em suma, a simulação computacional consolidou-se neste trabalho não como um fim em si mesma, mas como uma lente poderosa através da qual o engenheiro pode interpretar e otimizar a realidade. Espera-se que este material sirva como catalisador para que futuros profissionais projetem sistemas produtivos mais eficientes, resilientes e inteligentes.

6 REFERÊNCIAS

ANYLOGIC. The AnyLogic Company. Disponível em: <https://www.anylogic.com>. Acesso em: 2026.

BANKS, J.; CARSON, J. S.; NELSON, B. L.; NICOL, D. M. Discrete-Event System Simulation. 5. ed. Upper Saddle River: Pearson, 2010.

BONABEAU, E. Agent-based modeling: Methods and techniques for simulating human systems. Proceedings of the National Academy of Sciences, v. 99, n. suppl 3, p. 7280-7287, 2002.

BORSHCHEV, A. The Big Book of Simulation Modeling: Multimethod Modeling with AnyLogic 6. Chicago: AnyLogic North America, 2013.

FORRESTER, J. W. Industrial Dynamics. Cambridge: MIT Press, 1961.

GRIGORYEV, I. AnyLogic in Three Days: A Quick Course in Simulation Modeling. 5. ed. Chicago: AnyLogic North America, 2018.

KELTON, W. D.; SADOWSKI, R. P.; ZUPICK, N. B. Simulation with Arena. 6. ed. New York: McGraw-Hill Education, 2014.

LAW, A. M. Simulation Modeling and Analysis. 5. ed. New York: McGraw-Hill Education, 2015.

MACAL, C. M.; NORTH, M. J. Tutorial on agent-based modeling and simulation. Journal of Simulation, v. 4, n. 3, p. 151-162, 2010.

ROBINSON, S. Simulation: The Practice of Model Development and Use. 2. ed. Houndsmills: Palgrave Macmillan, 2014.

SENGE, P. M. The Fifth Discipline: The Art and Practice of the Learning Organization. New York: Doubleday/Currency, 1990.

STERMAN, J. D. Business Dynamics: Systems Thinking and Modeling for a Complex World. Boston: Irwin/McGraw-Hill, 2000.