

UNIVERSIDADE FEDERAL DE ITAJUBÁ

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO**

**ANÁLISE ELEMENTAR DA COMPLEXIDADE DE
MODELOS DE SIMULAÇÃO DISCRETOS E HÍBRIDOS**

João Paulo Barbieri

ITAJUBÁ

2023

UNIVERSIDADE FEDERAL DE ITAJUBÁ

**PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE PRODUÇÃO**

João Paulo Barbieri

**ANÁLISE ELEMENTAR DA COMPLEXIDADE DE
MODELOS DE SIMULAÇÃO DISCRETOS E HÍBRIDOS**

Tese submetida ao Programa de Pós-Graduação em Engenharia de Produção como parte dos requisitos para obtenção do título de Doutor em Engenharia de Produção.

Área: Engenharia de Produção

Orientador: Prof. Dr. Alexandre Ferreira de Pinho
Coorientador: Prof. Dr. José Arnaldo Barra Montevechi

ITAJUBÁ

2023

DEDICATÓRIA

Dedico este trabalho à Deus, à Jesus Cristo, à Maria Santíssima e a todos os santos! Também o dedico aos meus queridos pais, irmã, esposa, filhos, e a todas as pessoas que habitam em meu coração.

AGRADECIMENTOS

Tenha fé! Tenha fé quando tudo parece estar confuso. Tenha fé diante do tudo, diante do nada. Tenha fé quando os desafios parecem ser enormes. Tenha fé e não dobre os joelhos diante do medo. Tenha fé quando o cansaço abater o seu ânimo. Tenha fé mesmo nos momentos mais sombrios de sua vida. Deus, obrigado! Obrigado por estar presente nos detalhes mais singelos de minha vida. Obrigado por não me abandonar nos momentos mais solitários. Obrigado por me animar quando o ato de desistir me pareceu um bom caminho. Obrigado pela fortaleza, virtude que continuo aprendendo a cultivar.

Deus colocou em meu caminho pessoas importantes que me ajudaram no meu desenvolvimento espiritual e intelectual. Meus pais e a minha irmã são essas pessoas. Papai e mamãe, obrigado por me ensinarem o valor da honestidade, da tolerância, da determinação, do trabalho e do respeito ao próximo. Agradeço a você também irmã por estar presente em todos os momentos que precisei. A sua ajuda, suas palavras de incentivo e orações foram fundamentais.

Desde do início do doutorado, uma das pessoas que me ajudaram bastante foi a Flávia Motta. Flávia, agradeço muito pelo seu apoio, pelas suas palavras e orações. Flávio Vilela, suas palavras também me deram força para prosseguir, obrigado! Pedro Carvalho, agradeço pelas conversas, pelo apoio e por vibrar nos momentos decisivos dessa minha caminhada. Ju, agradeço imensamente seu carinho, amor e apoio; sem as suas palavras e orações, meus passos finais não seriam dados nesta reta final. De fato, muitas pessoas me apoiaram, e elas contribuíram com diversas palavras de incentivo. Mesmo não citando todas elas, deixo aqui os meus agradecimentos a todos que estiveram ao meu lado.

Não poderia deixar de agradecer a todos os professores que contribuíram para o meu desenvolvimento intelectual e acadêmico. De modo especial, gostaria de agradecer imensamente ao Prof. Dr. Alexandre Ferreira de Pinho e ao Prof. Dr. José Arnaldo Barra Montevechi pela paciência, e também por terem acreditado e confiado no meu trabalho. Obrigado por estarem ao meu lado nesta pesquisa e por terem me incentivado a continuar.

Concluindo, gostaria de agradecer a todos os servidores da Universidade Federal de Itajubá (UNIFEI), à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) pelo apoio e incentivo à pesquisa.

*“Pela educação a alma se liberta da prisão
subjetiva, do egocentrismo cognitivo próprio
da infância, e se abre para a grandeza e a
complexidade do real.”*

Olavo de Carvalho

RESUMO

Sob a influência do desenvolvimento tecnológico, os sistemas reais estão tendo um aumento expressivo quanto ao seu tamanho e à sua complexidade. Diante de um contexto de avanços tecnológicos, a simulação mantém sua relevância científica, servindo de apoio à tomada de decisões empresariais. Por meio da simulação, sistemas reais, e suas idiossincrasias, são analisados e aperfeiçoados. Durante um projeto de simulação, o especialista precisa tomar uma série de decisões, o que inclui a definição do nível de detalhamento do modelo. Além disso, na fase de desenvolvimento do modelo computacional, o especialista também precisa decidir sobre qual abordagem de simulação a ser utilizada. Em meio às decisões deste especialista, um dilema emerge: os sistemas reais estão progressivamente se tornando maiores e complexos em função do progresso tecnológico; mesmo com os avanços tecnológicos, a literatura científica afirma que um modelo computacional é uma abstração da realidade e ele deve ser o mais simples possível. No contexto deste dilema, a presente tese tem como objetivo geral aprofundar as discussões sobre a introdução de maior nível de detalhamento em modelos computacionais, considerando as abordagens Simulação a Eventos Discretos (SED) e Simulação Híbrida (SH), *id est*, abordagem SED combinada à abordagem Simulação Baseada em Agentes (SBA). A necessidade em aprofundar estas discussões gerou o método iDAV, método utilizado para mensurar modelos computacionais. Com a aplicação do método iDAV, constatou-se que os modelos discretos são mais simples de serem desenvolvidos quando o nível de detalhe é menor. Por outro lado, quando o escopo e o nível de detalhamento são aumentados, os modelos híbridos mostram-se mais adequados.

Palavras-chave: Simulação a Eventos Discretos; Simulação Baseada em Agentes; Simulação Híbrida; Complexidade; iDAV.

ABSTRACT

Under the influence of technological development, real systems are experiencing a significant increase in size and complexity. Faced with a context of technological advances, simulation maintains its scientific relevance, supporting business decision-making. Through simulation, real systems, and their idiosyncrasies, are analyzed and improved. During a simulation project, the specialist needs to make a series of decisions, which includes defining the model's level of detail. Furthermore, in the computational model development phase, the specialist also needs to decide on which simulation approach to use. In the midst of this specialist's decisions, a dilemma emerges: real systems are progressively becoming larger and more complex as a result of technological progress; even with technological advances, the scientific literature states that a computational model is an abstraction of reality and it should be as simple as possible. In the context of this dilemma, the present thesis has the general objective of deepening discussions on the introduction of a greater level of detail in computational models, considering the Discrete Event Simulation (DES) and Hybrid Simulation (HS) approaches, id est, DES approach combined with Agent-Based Simulation (SBA) approach. The need to deepen these discussions generated the iDAV method, a method used to measure computational models. With the application of the iDAV method, it was found that discrete models are simpler to be developed when the level of detail is lower. On the other hand, when the scope and level of detail are increased, hybrid models are more suitable.

Keywords: Discrete-Event Simulation; Agent-Based Simulation; Hybrid Simulation; Complexity; iDAV.

LISTA DE FIGURAS

Figura 2.1 - Série histórica de artigos sobre SH no WSC _____	19
Figura 2.2 - Resumo do levantamento bibliográfico _____	21
Figura 2.3 - Evolução temporal de publicações científicas sobre SBA e SED _____	22
Figura 2.4 - Categorização dos artigos por setor econômico _____	23
Figura 3.1 - Complexidade e precisão do modelo de simulação _____	40
Figura 4.1 - Técnica de desenvolvimento de um modelo de simulação _____	44
Figura 4.2 - Método iDAV _____	45
Figura 4.3 - Modelo conceitual do sistema de manufatura _____	47
Figura 4.4 - Diagrama de estado dos operadores dos modelos D0N3 e H0N3 _____	48
Figura 4.5 - Diagrama de estado dos operadores dos modelos D1N3 e H1N3 _____	49
Figura 4.6 - Procedimento estatístico para a validação dos modelos computacionais _____	52
Figura 5.1 - Grafo das linhas de código 182 a 199 _____	55
Figura 5.2 - Grafo de estrutura do modelo D0N3 _____	57
Figura 5.3 - Grafo de estrutura do modelo H0N3 _____	57
Figura 5.4 - Grafo de estrutura do modelo D1N3 _____	58
Figura 5.5 - Grafo de estrutura do modelo H1N3 _____	58
Figura 5.6 - Grafo de essência do modelo D0N3 _____	59
Figura 5.7 - Grafo de essência do modelo D0N3 _____	59
Figura 5.8 - Grafo de essência do modelo D1N3 _____	60
Figura 5.9 - Grafo de essência do modelo H1N3 _____	60
Figura 5.10 - Grafos de estrutura dos modelos D0N3 (a) e H0N3 (b) _____	65
Figura 5.11 - Grafos de estrutura dos modelos D1N3 (c) e H1N3 (d) _____	65
Figura 5.12 - Grafos de essência dos modelos D0N3 (e) e H0N3 (f) _____	66
Figura 5.13 - Grafos de essência dos modelos D1N3 (g) e H1N3 (h) _____	67
Figura 6.1 - Grafos de estrutura dos modelos D0N3 (a) e D0S3 (b) _____	70
Figura 6.2 - Grafos de estrutura dos modelos H0N3 (c) e H0S3 (d) _____	71
Figura 6.3 - Grafos de estrutura dos modelos D1N3 (e) e D1S3 (f) _____	71
Figura 6.4 - Grafos de estrutura dos modelos H0N3 (g) e H0S3 (h) _____	72
Figura 6.5 - Grafos de essência dos modelos D0N3 (i) e D0S3 (j) _____	73
Figura 6.6 - Grafos de essência dos modelos H0N3 (k) e H0S3 (l) _____	73
Figura 6.7 - Grafos de essência dos modelos D1N3 (m) e D1S3 (n) _____	74
Figura 6.8 - Grafos de essência dos modelos H1N3 (o) e H1S3 (p) _____	74

Figura 6.9 - Grafos de estrutura dos modelos D0N3 (a) e D0N9 (b)	76
Figura 6.10 - Grafos de estrutura dos modelos H0N3 (c) e H0N9 (d)	77
Figura 6.11 - Grafos de estrutura dos modelos D1N3 (e) e D1N9 (f)	77
Figura 6.12 - Grafos de estrutura dos modelos H1N3 (g) e H1N9 (h)	78
Figura 6.13 - Grafos de essência dos modelos D0N3 (i) e D0N9 (j)	78
Figura 6.14 - Grafos de essência dos modelos H0N3 (k) e H0N9 (l)	79
Figura 6.15 - Grafos de essência dos modelos D1N3 (m) e D1N9 (n)	79
Figura 6.16 - Grafos de essência dos modelos H1N3 (o) e H1N9 (p)	80
Figura 6.17 - Relação entre a densidade dos grafos e os modelos computacionais	81

LISTA DE QUADROS

Quadro 3.1 - Características dos modelos de SED e de SBA_____ 35

LISTA DE TABELAS

Tabela 4.1 - Estimativas de desempenho _____	50
Tabela 5.1 - Resultados do procedimento de contagem _____	53
Tabela 5.2 - Detalhamento dos pesos atribuídos às conexões de cada modelo _____	56
Tabela 5.3 - Grau médio ponderado dos modelos _____	61
Tabela 5.4 - Comparação entre modelos que utilizam a mesma abordagem de simulação __	62
Tabela 5.5 - Comparação entre modelos que utilizam abordagens de simulação diferentes _	62
Tabela 5.6 - Proporção dos tipos de conexões de cada modelo computacional _____	63
Tabela 5.7 - Detalhamento percentual dos pesos atribuídos às conexões de cada modelo __	63
Tabela 6.1 - Adição dos elementos 3D e a densidade dos grafos _____	75
Tabela 6.2 - Quantidade de componentes e conexões com a adição de elementos 3D _____	75
Tabela 6.3 - Adição De novos postos de trabalho e a densidade dos grafos _____	80

LISTA DE SIGLAS

CC – *Complexidade Ciclomática*

iDAV – *identificação, Definição, Análise e Verificação*

IDEF-SIM – *Integrated Definition Methods - Simulation*

LC – *Linha de Código*

MSED – *Modelo de Simulação a Eventos Discretos*

MSH – *Modelo de Simulação Híbrido*

PO – *Pesquisa Operacional*

SBA – *Simulação Baseada em Agentes*

SC – *Simulação Contínua*

SED – *Simulação a Eventos Discretos*

SH – *Simulação Híbrida*

SMC – *Simulação de Monte Carlo*

SD – *Sistemas Dinâmicos*

TFC – *Tamanho Funcional Cognitivo*

LISTA DE SÍMBOLOS

a – agente;

B – conjunto de comportamentos que o agente possui;

S – conjunto de atributos estáticos do agente;

D – conjunto de atributos dinâmicos do agente que são atualizados durante a simulação;

N – arredores (a vizinhança) do agente;

M – conjunto de mecanismos que atualizam o estado do agente durante a simulação;

$v(G)$ – número ciclomático;

e – quantidade de arestas do sistema;

n – número de vértices do sistema;

p – quantidade de blocos de códigos conectados do sistema;

D0N3 – Discreto, sem operadores autônomos, sem elementos 3D e com 3 postos de trabalho;

H0N3 – Híbrido, sem operadores autônomos, sem elementos 3D e com 3 postos de trabalho;

D1N3 – Discreto, com operadores autônomos, sem elementos 3D e com 3 postos de trabalho;

H1N3 – Híbrido, com operadores autônomos, sem elementos 3D e com 3 postos de trabalho;

D0S3 – Discreto, sem operadores autônomos, com elementos 3D e com 3 postos de trabalho;

H0S3 – Híbrido, sem operadores autônomos, com elementos 3D e com 3 postos de trabalho;

D1S3 – Discreto, com operadores autônomos, com elementos 3D e com 3 postos de trabalho;

H1S3 – Híbrido, com operadores autônomos, com elementos 3D e com 3 postos de trabalho;

D0N9 – Discreto, sem operadores autônomos, sem elementos 3D e com 9 postos de trabalho;

H0N9 – Híbrido, sem operadores autônomos, sem elementos 3D e com 9 postos de trabalho;

D1N9 – Discreto, com operadores autônomos, sem elementos 3D e com 9 postos de trabalho;

H1N9 – Híbrido, com operadores autônomos, sem elementos 3D e com 9 postos de trabalho.

SUMÁRIO

1. A INTRODUÇÃO	15
1.1. Objetivos	17
1.2. Contribuições Científicas	17
1.3. Estrutura da Tese	17
2. A ANÁLISE BIBLIOMÉTRICA	19
3. A FUNDAMENTAÇÃO TEÓRICA	26
3.1. Abordagem Tradicional e a Emergente	28
3.2. Complexidade do Modelo de Simulação	37
4. A METODOLOGIA DE PESQUISA	42
4.1. Modelagem e Simulação	42
4.2. Método iDAV	44
4.3. Modelos Computacionais	46
5. A APLICAÇÃO DO MÉTODO iDAV	53
5.1. <i>identificação</i>	53
5.2. <i>Definição</i>	54
5.3. <i>Análise</i>	56
5.4. <i>Verificação</i>	68
6. AS ANÁLISES DE SENSIBILIDADE	70
6.1. Adição de elementos 3D	70
6.2. Aumento dos postos de trabalho	76
7. AS CONCLUSÕES	82
7.1. Análises e Conclusões	82
7.2. Sugestões para Pesquisas Futuras	84
AS REFERÊNCIAS BIBLIOGRÁFICAS	86

1. A INTRODUÇÃO

Sendo uma das ferramentas com grande potencial de análise, a simulação é uma abordagem científica que auxilia na concepção, planejamento, análise e otimização de sistemas de manufatura (SHANNON, 1975; SCHÖNEMANN *et alii*, 2015). Isso faz com que a simulação seja uma ferramenta relevante, servindo de apoio à tomada de decisão sobre variáveis existentes em sistemas reais. Cabe salientar que, conforme Brailsford *et alii* (2019), esses sistemas possuem problemas de natureza complexa. Tal natureza, segundo Gunal (2019), é potencializada pelos avanços tecnológicos impulsionados pelos conceitos que abrangem a Indústria 4.0.

Nas fases iniciais de um projeto de simulação, o especialista em simulação precisa tomar uma série de decisões. Duas delas são (i) a definição do escopo do modelo e (ii) a definição do nível de detalhe. Segundo Robinson (2008), o escopo do modelo se refere à definição do limite do modelo ou da amplitude do sistema real incluída no projeto de simulação. O mesmo autor define o nível de detalhamento como a quantidade de detalhe a ser incluída para cada componente que pertence ao escopo do modelo. Neste processo de tomada de decisão sobre o escopo e o nível de detalhamento do modelo, o especialista realiza suposições e simplificações. Isso ocorre em função do acesso aos dados e informações que englobam o sistema real estudado e seus componentes.

Além disso, em um projeto de simulação, no qual há a fase de desenvolvimento do modelo computacional, o especialista também precisa decidir sobre qual abordagem de simulação usar. Essa decisão, como afirma Brailsford *et alii* (2019), ocorre em razão da preferência ou *expertise* do especialista quanto à abordagem de simulação. De acordo com Jahangirian *et alii* (2010) e Scheidegger *et alii* (2018), as abordagens Simulação a Eventos Discretos (SED), Sistemas Dinâmicos (SD) e Simulação Baseada em Agentes (SBA) são as mais utilizadas dentro da Pesquisa Operacional (PO) para resolver problemas no ambiente de negócios, sendo que a SED é a abordagem mais tradicional no meio científico.

Por mais de quarenta anos, a base da comunidade científica de simulação, que atua dentro da área de PO, tem sido a SED (SIEBERS *et alii*, 2010; SCHEIDEGGER *et alii*, 2018). Não é à toa que, para muitos pesquisadores da área de PO, a palavra simulação é tratada como sinônimo do termo SED (BRAILSFORD, 2014b). Porém, com o surgimento da SBA, discussões surgiram no campo científico acerca das potencialidades da SBA comparadas com as da SED, como pode ser visto nos trabalhos de Siebers *et alii* (2010) e Brailsford (2014b).

Essas discussões giram em torno de temas que abarcam a modelagem e simulação do comportamento, da interação, da decisão e da adaptação de entidades presentes em um sistema real. Tanto na SBA quanto na SED, o nível de detalhamento das entidades pode ser aprofundado e é por essa razão que existem discussões sobre qual abordagem de simulação o especialista deve utilizar para retratar de modo mais adequado a realidade.

Há ainda a abordagem Simulação Híbrida (SH). Brailsford *et alii* (2019) afirmam que a SH é a combinação entre dois ou mais abordagens de simulação. Nesta tese, a combinação entre as abordagens SED e SBA será objeto de estudo. Para Eldabi *et alii* (2019), a SH é um tópico de pesquisa que também vem ganhando atenção na comunidade científica pelo fato de permitir o surgimento de *insights* mais profundos a respeito dos sistemas reais. Tako *et alii* (2019) argumentam que a SH é uma abordagem relevante para modelar problemas e sistemas modernos caracterizados pelo alto volume de dados e informações, pelo número de *stakeholders* com opiniões subjetivas e conflitantes, e pelo alto grau de complexidade.

É consenso na comunidade científica que um modelo computacional é uma abstração da realidade, portanto é inviável inserir todas as variáveis que compõe o sistema real. Conforme Sargent (2014), um modelo deve ser parcimonioso e ter a capacidade de atender a um objetivo específico. O mesmo autor afirma ainda que o modelo de simulação é um modelo de caráter estrutural, portanto ele deve conter os principais componentes lógicos e causais que ocorrem no sistema real. Para Sarjoughian (2017), o objetivo básico de um modelo computacional consiste em possuir o menor tamanho possível e a mínima complexidade possível.

Sob a influência do desenvolvimento tecnológico, os sistemas reais estão tendo um aumento expressivo quanto ao seu tamanho e à sua complexidade, e isso vem desafiando a modelagem e a simulação de tais sistemas (CICIRELLI; FURFARO; NIGRO, 2011; GUNAL, 2019). Portanto, diante disso, um dilema emerge: os sistemas reais estão progressivamente se tornando maiores e complexos em função do progresso tecnológico; mesmo com os avanços tecnológicos, a literatura científica afirma que um modelo computacional é uma abstração da realidade e ele deve ser o mais simples possível. Há um ponto de equilíbrio? Este ponto de equilíbrio é alcançado tanto por meio da abordagem SED quanto pela abordagem SH? Quais são os efeitos ao introduzir maior nível de detalhamento em modelos computacionais discretos e híbridos? A abordagem SH é a mais apropriada quando se deseja aumentar o nível de detalhamento de um modelo computacional? As respostas para essas questões serão expostas no decorrer desta tese.

1.1. Objetivos

A presente tese tem como objetivo geral aprofundar as discussões sobre a introdução de maior nível de detalhamento em modelos computacionais, considerando as abordagens Simulação a Eventos Discretos (SED) e Simulação Híbrida (SH). Esse aprofundamento será feito a partir de análises que envolvem a complexidade estrutural dos modelos computacionais. Além disso, os objetivos específicos são:

- Elaborar um método que seja capaz de mensurar a complexidade estrutural de modelos de simulação;
- Analisar a complexidade estrutural de modelos computacionais a partir de conceitos da Teoria de Grafos.

1.2. Contribuições Científicas

Esta pesquisa pretende contribuir com discussões recentes sobre a aplicação das abordagens Simulação Híbrida (SH) e Simulação a Eventos Discretos (SED) para retratar sistemas reais que, em razão dos avanços tecnológicos, estão se tornando mais complexos. Observa-se que, na literatura, a simulação é, por vezes, tratada como arte e isso possui impacto direto no desenvolvimento de projetos de simulação. De modo específico, em um projeto de simulação, as decisões sobre o nível de detalhe a ser adotado e a abordagem de simulação a ser escolhida ocorrem de maneira subjetiva. Nesse sentido, independente das decisões do especialista em simulação, um dos resultados principais desta tese é a criação de um *método* que possui a função de auxiliar o especialista na mensuração da complexidade de seus modelos computacionais. A partir dessa mensuração, o especialista será capaz de avaliar o conjunto de componentes e conexões intrínseco ao modelo computacional que está sendo desenvolvido por ele. Ademais, o especialista será capaz de investigar os componentes e conexões mais relevantes de seu modelo computacional e identificar oportunidades de melhoria no seu projeto de simulação.

1.3. Estrutura da Tese

A tese está estruturada em 7 (sete) capítulos que, no sentido *lato*, abordam conceitos referentes a simulação, a modelos computacionais e a complexidade. De modo *stricto*, os próximos parágrafos retratam o conteúdo de cada capítulo.

No *Capítulo 1* é feita uma introdução sobre o tema em que a contextualização, as justificativas e os objetivos da tese são apresentados. As contribuições científicas esperadas e a estruturação da obra científica também fazem parte desse capítulo. No *Capítulo 2* está a análise bibliométrica sobre estudos científicos dos últimos anos que tratam da SH. Tal análise é importante porque revela informações sobre a produção científica acerca da SH, abordagem em foco nesta tese. Já no *Capítulo 3* encontra-se o embasamento teórico sobre as abordagens de simulação e os conceitos em torno da complexidade de modelos de simulação. Ainda nesse capítulo são tratados os conceitos e definições sobre Simulação Computacional, Simulação a Eventos Discretos, Simulação Baseada em Agentes, Simulação Híbrida e Complexidade. Na sequência, é abordado no *Capítulo 4* a metodologia científica utilizada nessa tese, o método iDAV, método desenvolvido para mensurar a complexidade dos modelos computacionais, e o desenvolvimento dos modelos computacionais utilizados nesta tese. No *Capítulo 5*, aplica-se o método iDAV, detalhando as etapas: identificação, Definição, Análise e Verificação. Após a aplicação do método iDAV, vem o *Capítulo 6* em que testes de sensibilidade são executados com o intuito de identificar os efeitos gerados com a adição de elementos 3D e com o aumento do escopo do modelo de simulação. No *Capítulo 7* estão as conclusões, as contribuições científicas atingidas e as sugestões para futuros trabalhos científicos. Por último, encontram-se as referências utilizadas nesta tese.

2. A ANÁLISE BIBLIOMÉTRICA

No decorrer dos séculos, os avanços tecnológicos modificaram os sistemas de produção da humanidade. Hoje, recursos e esforços estão sendo destinados para o desenvolvimento de pesquisas que envolvem os conceitos da Indústria 4.0. Esses conceitos se apoiam na integração dos sistemas produtivos por meio de nove pilares: Robôs Inteligentes, Manufatura Aditiva, Simulação, Integração Horizontal e Vertical, Internet das Coisas, *Big Data*, Computação em Nuvem, Segurança Cibernética e Realidade Aumentada. No pilar Simulação, a Simulação Híbrida (SH) tem um papel potencial.

A SH pode ser definida como a combinação entre duas ou mais das seguintes abordagens de simulação: Simulação a Eventos Discretos (SED), Sistemas Dinâmicos (SD) e Simulação Baseada em Agentes (SBA). Na última década, a SH cresceu significativamente em termos de popularidade. Em congressos, em particular o *Winter Simulation Conference* (WSC), o número de trabalhos sobre SH também cresceu nos últimos anos. Esse fato pode ser comprovado pela Figura 2.1.

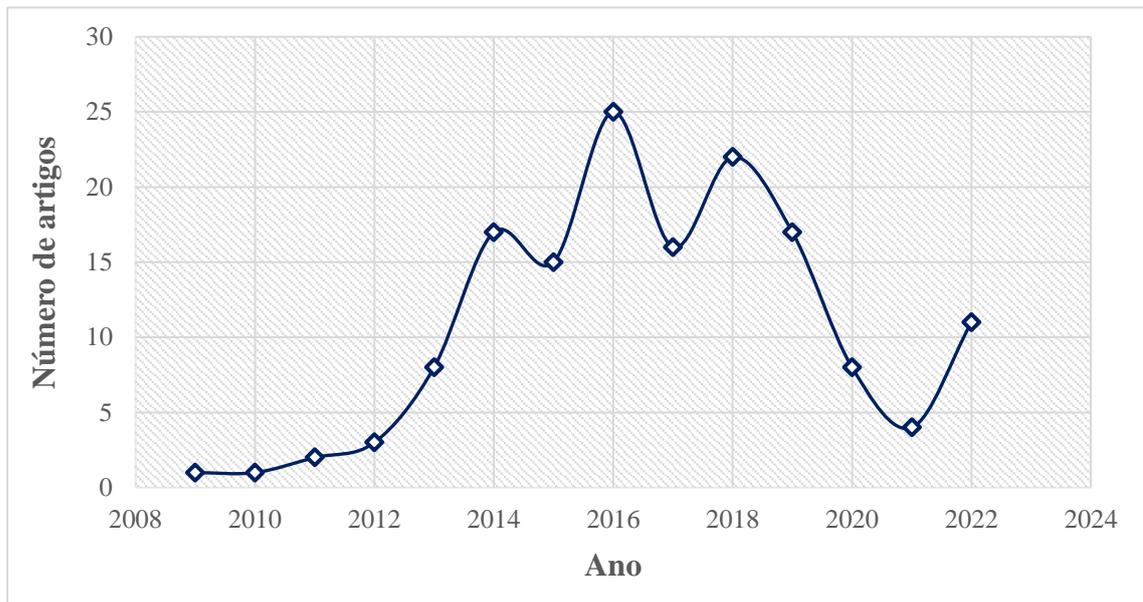


Figura 2.1 - Série histórica de artigos sobre SH no WSC

Fonte: Elaborado pelo próprio autor

O WSC é um dos principais congressos na área de simulação, conforme Scheidegger *et alii* (2018). Portanto, ele é um canal científico em que há a possibilidade de identificar temas emergentes na área de simulação. Devido ao aumento de trabalhos científicos envolvendo a SH, constata-se que a partir do ano 2014, nos anais do congresso, o WSC reservou uma seção específica para tratar da SH. Nota-se também que já existia um esforço em discutir a SH nos

anos 2012 e 2013 em seções como *Combined OR/Simulation Techniques*, *Application of Hybrid/Combined Simulation Techniques* e *Multi-Paradigm and Hybrid Simulation*.

A partir do levantamento bibliográfico, pode-se afirmar que há ausência de unidade em torno do termo SH. A pesquisa desenvolvida por Mustafee e Powell (2018) traz contribuições relevantes nesse sentido. Os autores fazem as seguintes distinções:

- ***Simulação Convencional (SC)***. Aplicação de uma abordagem de simulação na fase de implementação/desenvolvimento de modelo de simulação;
- ***Simulação Híbrida (SH)***. Aplicação de duas ou mais abordagens de simulação – *verbi gratia*: SBA, SED, SD – no estágio de implementação/desenvolvimento de modelo de simulação.
- ***Modelagem de Sistemas Híbridos (MSH)***. É a aplicação combinada de simulação com outras técnicas que vem de disciplinas como Computação Aplicada, Ciência da Computação, Engenharia de Sistemas e Pesquisa Operacional. A aplicação combinada não precisa necessariamente estar nos estágios de implementação/desenvolvimento de modelo computacional. Assim, essa aplicação combinada pode ocorrer, por exemplo, na fase de modelagem conceitual, na etapa de verificação e validação do modelo e nos estágios de experimentação.
- ***Estudo Híbrido de Modelagem e Simulação (EHMS)***. É a aplicação do MSH em uma ou mais etapas de um estudo de Modelagem e Simulação.

Portanto, considerando as fases tradicionais de um projeto de simulação – Concepção, Implementação e Análise –, a SC e SH se referem às abordagens de simulação escolhidas pelo especialista para a criação do modelo computacional na fase de Implementação. A MSH é mais flexível por admitir a combinação e aplicação de outras técnicas, oriundas de outras disciplinas, em uma das fases do projeto de simulação. Já o EHMS possui um escopo mais abrangente de cunho teórico e depende da aplicação da MSH.

Com base na literatura, o interesse particular pela SH pode ser explicado por dois fatores principais: (i) a progressiva inserção da tecnologia nos sistemas produtivos que contribui para o aumento de sua complexidade; e (ii) a possibilidade em modelar e simular aspectos de sistemas reais que não podem ser retratados por meio da utilização de apenas uma abordagem de simulação (MYKONIATIS; ANGELOPOULOU, 2020; BRAILSFORD *et alii*, 2019; ELDABI *et alii* 2019; GUNAL, 2019; CICIRELLI; FURFARO; NIGRO, 2011).

De acordo com Brailsford *et alii* (2019), a SD e SED é a combinação mais popular encontrada na literatura científica. Todavia, os mesmos autores notaram que, nos últimos anos, houve um crescimento da aplicação de conceitos da SBA nos modelos computacionais. Segundo os autores, esse crescimento é explicado por três fatores: (i) o aumento da utilização da SBA de modo geral; (ii) a crescente popularidade da área de Pesquisa Operacional Comportamental; (iii) a necessidade em incorporar nos modelos computacionais a tomada de decisões por entidades inteligentes/emocionais. Portanto, a partir da combinação SBA e SED, o sistema é modelado seguindo a abordagem *bottom-up*. Na combinação SED e SD, o sistema é modelado sob a perspectiva *top-down*.

Em razão dos três fatores supracitados e das discussões em torno da inclusão entidades inteligentes/emocionais nos modelos computacionais, foi feita uma revisão da literatura utilizando o banco de dados de citações Scopus®. A escolha desse banco de dados está embasada nos trabalhos desenvolvidos por Jahangirian *et alii* (2010) e Franceschini, Maisano e Mastrogiacomio (2014). Assim, foi utilizado a seguinte *string* de consulta: TITLE-ABS-KEY ("HYBRID SIMULATION" OR "HYBRID MODEL*") AND TITLE-ABS-KEY ("DISCRETE-EVENT" OR "DISCRETE EVENT") AND TITLE-ABS-KEY ("AGENT-BASED" OR "AGENT BASED"). O resumo dos tipos de documentos científicos encontrados nessa consulta encontra-se na Figura 2.2.

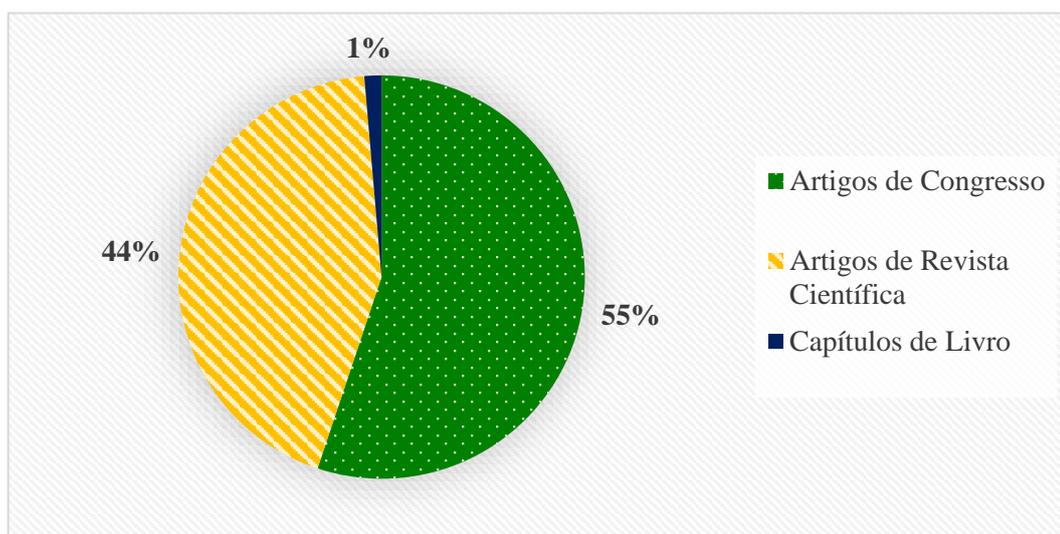


Figura 2.2 - Resumo do levantamento bibliográfico

Fonte: Elaborado pelo próprio autor

Na pesquisa foram encontrados 152 (cento e cinquenta e dois) documentos dos quais 5 (cinco) eram divulgação de congresso. Portanto, no total foram 147 (cento e quarenta e sete) trabalhos científicos tratando sobre SBA e SED. Nota-se que o volume de artigos de congresso é

superior ao volume de artigos publicados em revistas científicas. Em suma, foram encontrados 81 (oitenta e um) artigos publicados em congressos, 64 (sessenta e quatro) artigos publicados em revistas científicas e 2 capítulos de livro. O nome dos livros em que se encontram estes capítulos são *Discrete-Event Simulation and System Dynamics for Management Decision Making* – publicado na Grã-Bretanha pela organização John Wiley & Sons em 2014 –, e *New Trends in Emerging Complex Real Life Problems* – publicado na Itália pela organização Springer em 2018.

Do total de artigos de congresso, aproximadamente 23% são artigos científicos que se encontram nos anais do WSC e o restante pertence a outros congressos internacionais. Esse fato mostra a importância do WSC como meio de divulgação científica sobre os temas SBA e SED. Quanto aos artigos de revistas científicas, não há apenas uma revista que se destaca das demais quando o assunto é SBA e SED. A Figura 2.3 mostra a evolução temporal das publicações científicas sobre SBA e SED, tanto em congressos quanto em revistas científicas.

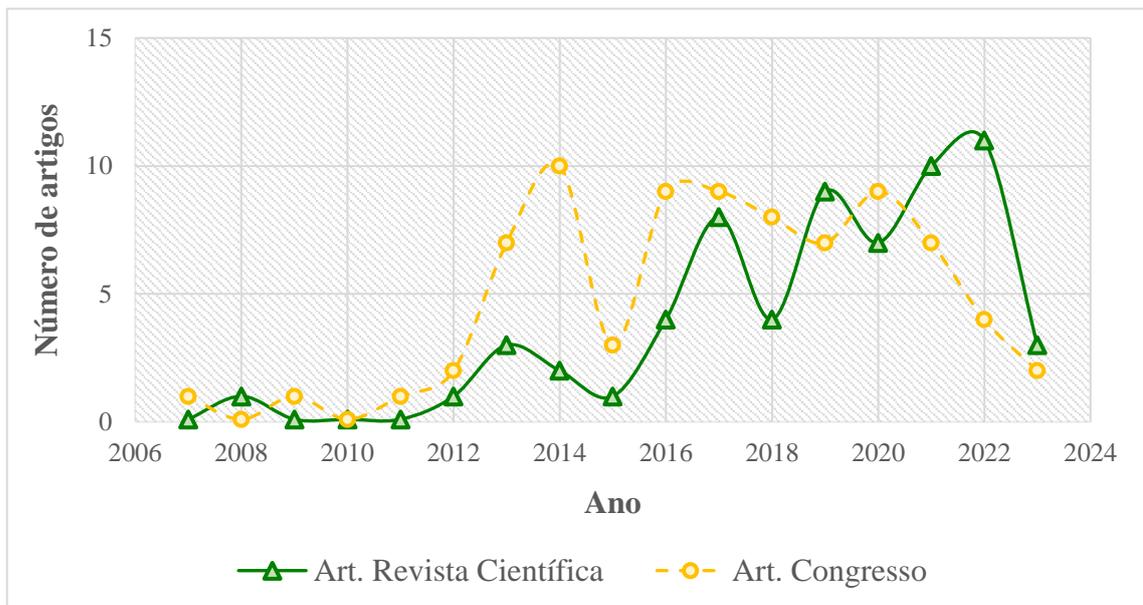


Figura 2.3 - Evolução temporal de publicações científicas sobre SBA e SED

Fonte: Elaborado pelo próprio autor

Tanto os artigos de congresso quanto os de revista científica foram filtrados com base na seguinte pergunta: o documento científico aplica a combinação das abordagens SBA e SED? Esse filtro é necessário para fazer três distinções: (i) artigos que pertencem a MSH, (ii) artigos que tratam da combinação SBA+SED+SD e (iii) artigos que tratam apenas da combinação SBA+SED. Do total de artigos, 54% tratam efetivamente da combinação SBA+SED, 34% abordam a combinação SBA+SED+SD e 12% representam os artigos que aplicam outras técnicas com as abordagens SBA, SED e SD.

Após essa filtragem, os artigos que combinam as abordagens SBA e SED foram categorizados conforme os três setores da economia: (i) Setor Primário, (ii) Setor Secundário e (iii) Setor Terciário. Tal categorização foi feita de modo a compreender a proporção de trabalhos científicos produzidos em cada setor econômico. De acordo com Figura 2.4, há uma proporção considerável de artigos desenvolvidos no terceiro setor econômico, no qual inclui empresas de comércio, transporte, prestação de serviços e saúde. Existem poucos trabalhos científicos voltados para o ramo agropecuário.

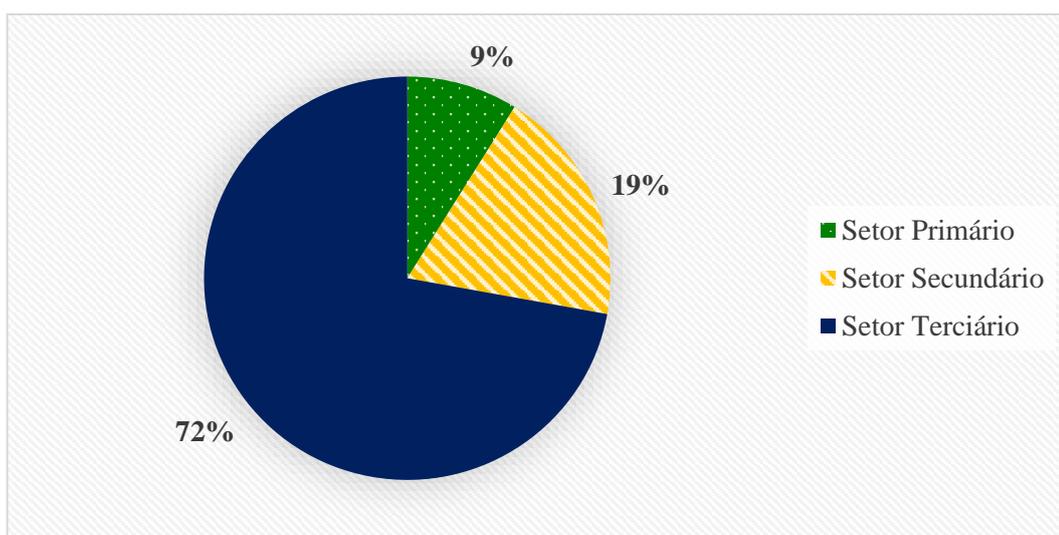


Figura 2.4 - Categorização dos artigos por setor econômico

Fonte: Elaborado pelo próprio autor

Em razão do foco dado nesta tese aos sistemas de manufatura, cabe destacar aqui os artigos mais recentes do segundo setor econômico que possuem a combinação SBA+SED. Estes artigos são: (FANI *et alii*, 2022), (RODRIGUES; PINHO; SENA, 2019), (NAGADI *et alii*, 2018), (SENA *et alii*, 2017) e (OLIVEIRA *et alii*, 2017).

Fani *et alii* (2022) desenvolveram um artigo no qual fica claro a preocupação com a sustentabilidade. De modo específicos, os autores aplicaram os conceitos da SH para desenvolver um modelo computacional que auxilia à tomada de decisão de *stakeholders* que atuam da indústria da moda. O modelo abrange o comportamento do consumidor dentro da cadeia produtiva, focando no consumo de itens para vestuário.

O artigo de Rodrigues, Pinho e Sena (2019) faz uso da SH de modo pragmático para estudar a programação de produção de um sistema do tipo *job shop*. O modelo computacional desenvolvido pelos autores é hipotético, porém teve como base sistemas reais de manufatura encontrados na literatura. Conforme os autores, os sistemas do tipo *job shop* são relevantes para a indústria que busca ter maior flexibilidade de *mix* e de volume de produção dentro do

mercado. Sobre o modelo híbrido do artigo, o sistema *job shop* foi retratado pela abordagem SED e o sequenciamento da produção foi definido a partir da abordagem SBA. As ordens de produção foram sequenciadas pelos agentes de modo a tornar o sistema *job shop* mais eficiente, diminuindo o *makespan*. O artigo possui contribuições científicas no campo da programação da produção de um sistema do tipo *job shop* e da SH ao apresentar um modelo computacional com agentes autônomos que tomam decisões de sequenciamento.

O trabalho científico de Nagadi *et alii* (2018) utiliza a SH para ajudar os tomadores de decisão a projetar melhores configurações e recursos em sistemas inteligentes de manufatura. Os autores desenvolveram um modelo que gera e avalia a pré-implementação de configurações ideais de manufatura. Além disso, o modelo fornece orientação para o estabelecimento de protocolos de mensagens apropriadas entre os componentes do sistema de manufatura. A abordagem SED foi utilizada para construir o sistema de manufatura do tipo *flow shop*. Já a abordagem SBA foi aplicada de maneira a retratar os estados de cada máquina presente no sistema, considerando, por exemplo, momentos de funcionamento, falha e manutenção.

Sena *et alii* (2017) desenvolveram um modelo híbrido que simula diversas alternativas para a alocação dinâmica de recursos humanos em um sistema de produção de alimentos. Com esse modelo, o objetivo foi o de aumentar o valor médio da margem da contribuição total. No decorrer do artigo, percebe-se que esse objetivo foi alcançado. No modelo computacional, o sistema produtivo foi retratado usando a abordagem SED. As decisões envolvendo a alocação dos operadores no sistema foram tomadas a partir da aplicação da abordagem SBA.

O quinto e último artigo é o de Oliveira *et alii* (2017). Nesse artigo os autores desenvolveram dois modelos híbridos que retratam dois sistemas de manufatura. Ambos os sistemas são do tipo *flow shop*. O objetivo central do artigo reside na discussão sobre a representação do fator humano em modelos computacionais. Tal representação ocorreu com a inclusão do ritmo circadiano de trabalho dos operadores por meio da abordagem SBA. Assim como nos artigos tratados anteriormente, os sistemas de manufatura foram construídos através da abordagem SED.

Em resumo, a partir do levantamento bibliográfico, constatou-se que nos últimos anos houve um crescimento de publicações sobre a SH, existem mais publicações sobre SH em congressos científicos – sobretudo no WSC –, e, a maior parte desses trabalhos científicos, estão sendo aplicados no terceiro setor econômico que envolve atividades de comércio de bens e prestação de serviços. As principais razões para o uso da SH encontram bases em argumentos relacionados com inserção da tecnologia em arranjos produtivos – o que eleva,

consequentemente, a complexidade dos sistemas de produção –, e a dificuldade em modelar e simular características desses sistemas apenas por meio de uma abordagem de simulação.

3. A FUNDAMENTAÇÃO TEÓRICA

Em diversos ramos da ciência, o progresso científico se deu com base em duas metodologias: (i) análise teórica ou dedução e (ii) análise empírica ou indução (HARRISON *et alii*, 2007). Inicialmente discutida como forma de raciocínio pelos gregos, especialmente por Platão, a dedução consiste em um processo em que conclusões, fundamentadas em argumentos teóricos ou lógicos e em um conjunto de premissas iniciais, são retiradas com relação a um fenômeno ou comportamento (BHATTACHERJEE, 2012; WALLIMAN, 2011).

A indução, sendo a primeira e a mais usual forma de atividade científica, foi a abordagem que deu sustentação à revolução científica no século XVII. Diferentemente do processo de dedução, conclusões são estabelecidas durante o processo de indução a partir dos fatos ou das provas observadas. Em outras palavras, no método indutivo, observações a respeito das variáveis (dados) são feitas e, posteriormente, essas observações são analisadas com o objetivo de encontrar relações entre essas variáveis (BHATTACHERJEE, 2012; HARRISON *et alii*, 2007; WALLIMAN, 2011).

Apesar das distinções existentes entre a simulação e as metodologias de dedução e de indução, há características análogas entre as abordagens. A simulação é semelhante à dedução pelo fato de que os resultados partem de um conjunto de premissas estabelecidas. A simulação também se assemelha a indução, pois ela possibilita inferências sobre as relações entre as variáveis através da análise dos dados de saída, mesmo que esses dados de saída sejam produzidos por programas de simulação (HARRISON *et alii*, 2007). Com o conhecimento das metodologias científicas, e tendo como referência os fatos históricos, a história da simulação e seu amadurecimento como ciência ficam mais compreensíveis.

A história da simulação pode ser abordada sob diferentes pontos de vista. *Verbi gratia*, sob a perspectiva de sua utilização para fins de análise, treinamento ou pesquisa; sob o ponto de vista dos tipos de abordagens de simulação existentes – Simulação a Eventos Discretos (SED), Simulação Contínua (SC), Simulação Baseada em Agentes (SBA) –; sob a ótica das linguagens de simulação ou ambientes de programação como GPSS®, SLAM®, Arena®, AutoMod®, Simio®, SIMSCRIPT®, SIMULA®; e sob o enfoque dos campos nos quais a simulação pode ser aplicada tais como a área militar e as áreas de manufatura, comunicações e transportes (GOLDSMAN; NANCE; WILSON, 2010).

Segundo Nance e Sargent (2002), o período da Segunda Guerra Mundial (1939 - 1945) foi o pano de fundo histórico em que a simulação teve início. Foi nessa época em que modelos

contínuos e de Monte Carlo foram desenvolvidos. Provavelmente, a SED nasceu durante a década de 1940; todavia, não há como afirmar a data correta conforme os autores. Quanto a SBA, Chwif e Medina (2015) afirmam que ela se consolidou efetivamente na década de 1990, apesar da definição de sua base teórica ter ocorrido no final da década de 1940.

Nas décadas posteriores ao período da Segunda Guerra Mundial, a simulação ganhou uma aceitação mais abrangente da comunidade científica e começou a ser aplicada em diversos estudos das áreas de ciências naturais e, seguidamente, das áreas de ciências sociais (HARRISON *et alii*, 2007).

Há uma relação intrínseca entre a evolução tecnológica e o desenvolvimento científico na área de simulação. Durante a década de 1950 houve um aumento na disponibilidade de computadores eletrônicos para propósitos diversos. Esse fato tornou propício a rápida difusão das técnicas de simulação e suas aplicações (GOLDSMAN; NANCE; WILSON, 2010). Um dos fatores que influenciou esse aumento no uso das técnicas de simulação, especialmente na área de manufatura, foram o aperfeiçoamento das capacidades de hardware e o aprimoramento dos softwares de simulação (GIANNASI; LOVETT; GODWIN, 2001).

No que se refere aos avanços na área de software e a relação desses avanços com a simulação, a emergência da linguagem de programação FORTRAN® ampliou, no final dos anos de 1950, a quantidade de semânticas úteis para a compreensão das representações de modelo. Em meados dos anos de 1960, surgiram a GASP® e a MILITRAN® (pacotes de simulação FORTRAN®) oferecendo capacidades funcionais, inibindo a difusão da *Simulation Programming Language* (SPL) –, um dos primeiros programas de simulação para construção de modelos. Em paralelo, a linguagem ALGOL® tinha ampla aceitação no continente europeu. A partir dessa linguagem surgiu a SIMULA 67®, inaugurando o estilo de programação orientada a objeto (NANCE; SARGENT, 2002).

Desde os anos de 1980, a evolução da capacidade de muitos softwares de simulação acompanhou o crescimento do poder computacional (BRAILSFORD; HARPER; SYKES, 2012). Nas décadas de 1980 e 1990, a linguagem orientada a objeto se tornou popular através da Smalltalk e se estabeleceu como uma metodologia de software (NANCE; SARGENT, 2002).

Ressalta-se que, segundo Fioretti (2012), a programação orientada a objeto proporcionou novas oportunidades de desenvolver simulações por meio do computador. Fazendo uso da

linguagem orientada a objeto, a representação dos atores do mundo real pode ocorrer naturalmente por meio de objetos.

Atualmente, com os avanços tecnológicos e as expectativas geradas em torno dos conceitos de Indústria 4.0, a simulação mantém seu espaço de relevância no meio empresarial em razão da oportunidade de desenvolvimento de gêmeos digitais. Nesse contexto, todo o *know-how* construído no campo científico da simulação se torna importante para compreender a complexidade dos arranjos empresariais.

Nas próximas seções, conceitos e definições envolvendo as abordagens SED, SBA e Simulação Híbrida (SH) serão descritos. Em razão dos objetivos traçados nesta tese, uma seção será dedicada ao tema complexidade do modelo de simulação.

3.1. Abordagem Tradicional e a Emergente

A simulação, também chamada de modelagem e simulação, é uma das técnicas para estudar um determinado sistema. De acordo com Shannon (1975), a modelagem e simulação é, dentre outras ferramentas de análise, aquela com maior poder, permitindo ao especialista a realização de experimentos com modelos que retratam sistemas reais. Sendo influenciada pelas áreas de matemática, de ciência da computação, de probabilidade e de estatística, a simulação possui um conceito simples e, intuitivamente, atraente.

Há na literatura diversas definições para simulação. Shannon (1975, p. 2, tradução nossa) define simulação da seguinte forma:

Simulação é o processo de formular um modelo de um sistema real e de realizar experiências com este modelo como o propósito ou de compreender o comportamento do sistema ou de avaliar várias estratégias (dentro dos limites impostos por um ou mais critérios) para a operação do sistema.

Para Baines *et alii* (2004), simulação consiste em uma técnica que possibilita a construção de um modelo capaz de descrever o comportamento de um sistema do mundo real. A partir desse modelo, testes podem ser efetuados para verificar como o desempenho do sistema proposto reage sob diferentes condições de funcionamento.

Ingalls (2008) estabelece que, possuindo a finalidade de avaliar diferentes estratégias de operação ou compreender o comportamento do sistema, a simulação baseia-se na execução do processo de criação de um modelo dinâmico a partir de um sistema real dinâmico. Logo, é possível imitar o comportamento de um sistema dinâmico através da simulação.

Sturrock (2014) completa essas definições sobre simulação quando afirma que simulação consiste em um processo de descoberta, uma vez que conhecimentos são obtidos desde o esforço em descrever com precisão o sistema real até o alcance dos primeiros resultados gerados pela simulação.

Simulação é um meio que possibilita analisar os fenômenos de um sistema real ao manipular as principais variáveis desse sistema em um ambiente virtual. A simulação possui limitações, não apenas de ordem computacional, mas também de metodológica quanto à transformação do sistema real em um modelo abstrato.

Na literatura científica, existem abordagens de simulação que possuem características diferentes. O especialista em simulação que busca transformar um sistema real em um modelo computacional pode utilizar as seguintes abordagens de simulação:

- **Simulação de Monte Carlo (SMC)**. Também conhecida como simulação estática, a SMC faz uso de geradores de números aleatórios e de distribuições de probabilidade acumulativas do sistema de interesse. Uma roleta, uma tabela de dígitos aleatórios, uma sub-rotina de computador ou qualquer outro método em que é possível se obter números aleatórios distribuídos uniformemente pode ser utilizado como gerador de números aleatórios. Quanto à distribuição de probabilidade, ela pode se originar de dados empíricos de registros anteriores, de uma experiência recente ou de uma distribuição teórica já conhecida. Sendo particularmente útil na resolução de problemas matemáticos complexos, neste tipo de simulação, o elemento tempo não é relevante. Logo, o sistema simulado através da SMC é representado em um dado instante do tempo (SHANNON, 1975; CHWIF; MEDINA, 2015).
- **Simulação Contínua (SC)**. A SC é aplicada quando há a necessidade de modelar um sistema que muda continuamente as suas variáveis de estado ao longo do tempo. Ou seja, a SC é apropriada quando o sistema de interesse possui um fluxo contínuo de informações ou de componentes. É comum, nesse tipo de simulação, o uso de equações diferenciais para calcular as alterações das variáveis de estados (SHANNON, 1975; LAW, 2007; CHWIF; MEDINA, 2015).
- **Sistemas Dinâmicos (SD)**. Uma das principais características da SD é o foco nas relações de causa e efeito entre os elementos do sistema a ser simulado. Assim, sendo extremamente útil para identificar as variáveis de um sistema e suas relações causais, a SD também é útil para compreender de modo global o comportamento de um sistema.

Outras características importantes desse tipo de simulação são a retroalimentação e o uso de uma abordagem *top-down* (de cima para baixo). É importante afirmar que, nos modelos de SD, o elemento tempo é um parâmetro relevante quando o comportamento do sistema é considerado. Além disso, os modelos de SD são deterministas e não conseguem capturar a variabilidade individual dos elementos do sistema (MACAL; NORTH, 2005; VIANA *et alii*, 2014; CHWIF; MEDINA, 2015).

- ***Simulação a Eventos Discretos (SED)***. Por meio de eventos, um determinado sistema sofre mudanças de estado em momentos discretos do tempo. O sistema que possui essas características pode ser modelado através da SED. Nesse tipo de simulação também se considera as alterações no estado do sistema ao longo do tempo. Um modelo de simulação a eventos discretos é geralmente orientado através do tempo com base em amostras aleatórias de distribuições de probabilidade. Dessa maneira, esse tipo de simulação é normalmente de natureza estocástica (LAW, 1988; CHWIF; MEDINA, 2015).
- ***Simulação Baseada em Agentes (SBA)***. Na SBA, a construção de um modelo baseado em agentes ocorre a partir da modelagem dos agentes do sistema que possuem características autônomas. Além dessas características autônomas, esses agentes têm capacidades interativas, ou seja, eles podem interagir entre si e com o ambiente — mundo virtual de caráter discreto ou contínuo. Através do uso da SBA é possível modelar de modo explícito a complexidade que advém das ações individuais (SIEBERS *et alii*, 2010; CHWIF; MEDINA, 2015).

Dentre as abordagens de simulação apresentadas, a SD e a SED são consideradas as mais tradicionais no meio científico. Já a SBA é a abordagem de simulação mais recente (LAW, 2015; SCHEIDEGGER *et alii*, 2018).

A SED sempre esteve, historicamente, muito próxima da Pesquisa Operacional (PO). Há mais de 40 anos que a SED vem sendo o principal pilar da comunidade científica que desenvolve trabalhos envolvendo a simulação de sistemas e processos dentro da PO (INGALLS, 2008; SIEBERS *et alii*, 2010; SCHEIDEGGER *et alii*, 2018).

Além dessa ligação com a PO, a SED encontra-se bem próxima da Ciência da Computação sob a ótica do desenvolvimento de softwares e algoritmos (BRAILSFORD, 2014b). Nos últimos anos, apesar de ser uma ferramenta computacionalmente cara, o uso da SED aumentou devido à dois fatores: o aumento do poder computacional e o aumento de memória

computacional (NEGAHBAN; SMITH, 2014). Hoje, softwares sofisticados de SED podem proporcionar ao usuário meios interativos de gerar análises de dados de saída do modelo (FAKHIMI *et alii*, 2014). Além disso, existem softwares que permitem a criação de modelos de simulação no formato tridimensional.

Com a SED é possível lidar com incertezas estocásticas e ter um entendimento acerca do processo que está presente no sistema (MACAL; NORTH, 2005). Assim, de acordo com Siebers *et alii* (2010), é adequado utilizar a SED em problemas que envolvem simulações de filas ou redes de filas complexas. Nesses casos, os processos podem ser bem definidos e as incertezas desses processos podem ser representadas por meio de distribuições estocásticas.

Portanto, possuindo a habilidade de imitar a dinâmica de sistemas reais, a SED é uma ferramenta que permite modelar sistemas que possuem filas e atividades. Uma das características mais acentuadas desses sistemas é a mudança de estado em momentos discretos do tempo (INGALLS, 2008; KASAIE; KELTON, 2015).

Conceitualmente, Law (2015) afirma que a SED se preocupa em modelar sistemas que sofrem alterações com o passar do tempo. Essas alterações ocorrem instantaneamente em pontos distintos no tempo em virtude das variáveis de estado do sistema. São nesses pontos distintos no tempo que eventos surgem. Nesse contexto, um evento pode ser definido como um acontecimento instantâneo capaz de mudar o estado do sistema. É verdade que a SED pode ser realizada através de cálculos manuais, todavia, por conta do volume de dados característicos dos sistemas reais que precisam ser manipulados e armazenados, é comum que esse tipo de simulação seja realizado por meio de ferramentas computacionais.

A SED é aplicada em diferentes áreas, *verbi gratia*: Engenharia (PERGHER *et alii*, 2020), Ciência da Computação (BARBIERATO; GRIBAUDO; SERAZZI, 2020), Matemática (AZEVEDO *et alii*, 2020), Medicina (MARSHALL *et alii*, 2023), Gestão (KAMBLI; SINHA; SRINIVAS, 2020), Ciência de Tomada à Decisão (GUL; GUNERI; GUNAL, 2020), Física e Astronomia (RAEDT *et alii*, 2020), Ciência de Materiais (ZAKIKHANI; NASIRI; ZAYED, 2020), Ciências Sociais (ABEL; FAUST, 2020) e Ciência Ambiental (CHEN *et alii*, 2020). Conforme Siebers *et alii* (2010), no contexto organizacional existem muitas aplicações para a SED em áreas como manufatura e serviços. Chwif e Medina (2015) afirmam que é ampla a aplicação da SED nessas duas áreas.

Como já afirmado, a SED é uma das abordagens de simulação mais tradicionais no meio científico e, portanto, são diversas as aplicações, principalmente nas áreas de manufatura e de

serviços. No entanto, uma abordagem de simulação emergente vem ganhando espaço no cenário científico. Esta abordagem de simulação é a SBA.

A SBA é uma abordagem relativamente nova. A sua popularidade vem crescendo assim como seu o número de aplicações em diversas áreas, por exemplo: Ciência da Computação (TOLL; PETTRÉ, 2020), Ciências Sociais (HWANG, 2020), Engenharia (VIZZARI; CROCIANI; BANDINI, 2020), Matemática (DELCEA, 2020), Ciência Ambiental (CHUDZINSKA *et alii*, 2020), Agricultura e Ciências Biológicas (FERNANDEZ-MENA *et alii*, 2020), Bioquímica, Genética e Biologia Molecular (OZALP *et alii*, 2020), Economia, Econometria e Finanças (ALEXANDRE; LIMA, 2020), Negócios, Gestão e Contabilidade (VINOGRADOV; LEICK; KIVEDAL, 2020), Medicina (POSSIK *et alii*, 2023).

Cientistas da área de computação, de economia, de biologia e de sociologia foram os que conduziram o desenvolvimento e a utilização da SBA; cada qual de acordo com suas próprias metas e objetivos específicos (SIEBERS *et alii*, 2010). A ciência da complexidade, a ciência de sistemas, a ciência da computação, a ciência de gestão, as ciências sociais em geral, a área de sistemas dinâmicos e a área de modelagem e simulação são os campos científicos que sustentam os fundamentos teóricos, a visão de mundo conceitual, a filosofia e as técnicas de modelagem da SBA (MACAL; NORTH, 2005; MACAL; NORTH, 2009).

A SBA é uma abordagem flexível e poderosa para modelar comportamentos dinâmicos e detalhados presentes em sistemas complexos. Também há diversos relatos de sucesso da aplicação da SBA no contexto industrial (MORTAZAVI; KHAMSEH; AZIMI, 2015).

Os avanços na área de SBA sugerem impactos de longo alcance quanto a forma com que as organizações utilizam os computadores como ferramentas de auxílio a tomada de decisão, e quanto a maneira com que os modelos de SBA são usados pelos pesquisadores para atingir descobertas na área científica. Portanto, fica visível a existência de elevado interesse em desenvolver modelos através da SBA que tenham aplicação em diversos problemas (MACAL; NORTH; SAMUELSON, 2013).

Por ser uma área de pesquisa relativamente nova, o termo *agente* gera discussões no meio acadêmico. Apesar dessa não concordância, existem pontos em comum entre as definições (MACAL; NORTH, 2005). De acordo com Mortazavi, Khamseh e Azimi (2015), agentes são elementos autônomos que se auto organizam através de regras específicas de tomada de decisões. Para Brailsford (2014a), agentes são indivíduos presentes na SBA. Esses indivíduos são autônomos, agem de modo independente no modelo, podem se lembrar de forma perfeita

ou imperfeita sobre o que ocorreu no passado, podem possuir a capacidade de aprender com o passado e a capacidade de adaptar suas reações e comportamentos em um cenário futuro. Os agentes também são capazes de se comunicarem uns com os outros e com o próprio ambiente.

Visando uma modelagem mais prática, Macal e North (2014) afirmam que os agentes possuem as seguintes propriedades e atributos:

- **Autonomia.** A autonomia é uma característica própria de um agente. O comportamento de um agente consiste em um processo em que as informações são captadas do ambiente, e das ações e decisões do próprio agente. Partindo de uma série limitada de situações emergentes, inerentes ao modelo, um agente pode atuar de maneira independente de seu ambiente e de suas interações com outros agentes.
- **Modularidade.** Os agentes seguem o princípio da modularidade. Um agente é um elemento discreto e identificável com a capacidade de efetuar decisões. Ele possui um conjunto de atributos e comportamentos. A condição de modularidade exige que um agente possua um limite. É através desse limite que se torna possível discernir se algo no estado do modelo faz parte ou não de um agente, ou se isso é uma particularidade compartilhada entre os agentes do modelo.
- **Sociabilidade.** Um dos aspectos de um agente é a sociabilidade. Assim, os agentes se interagem uns com os outros. Há algumas regras comuns de interação que englobam questões como a disputa por espaço, a prevenção de colisões, o reconhecimento do agente, a troca de informações, a influência, e outros mecanismos específicos.
- **Condicionalidade.** Um agente possui um estado e esse estado sofre variações ao longo do tempo. Um conjunto ou subconjunto dos atributos e dos comportamentos do agente reflete o estado desse agente. O estado de um agente representa a sua condição que, por sua vez, é definida por variáveis essenciais que se encontram associadas ao estado atual desse agente. Os estados de todos os agentes e o estado do ambiente compõem o estado de um modelo de SBA. É importante acrescentar que estão condicionados no estado de um agente seus comportamentos. Portanto, quanto mais requintado é o conjunto de comportamentos de um agente, mais rico é o conjunto de possíveis estados desse agente.

Matematicamente, Macal, North e Samuelson (2013) ainda afirmam que um agente pode ser retratado através de cinco elementos essenciais como mostra a Equação 3.1.

$$\mathbf{a} = \{\mathbf{B}, \mathbf{S}, \mathbf{D}, \mathbf{N}, \mathbf{M}\} \quad (3.1)$$

A letra **B** representa um conjunto de comportamentos que o agente possui; sobre esses comportamentos, eles equivalem, por exemplo, às ações de decidir, comunicar, movimentar e aprender. A letra **S** é o conjunto de atributos estáticos do agente; variáveis demográficas como nome, data de nascimento, sexo podem ser atributos estáticos. Por outro lado, a letra **D** reflete o conjunto de atributos dinâmicos do agente que são atualizados durante a simulação; a memória e as interações do agente são exemplos de atributos dinâmicos. **N** simboliza os arredores (a vizinhança) do agente; esses arredores demarcam os espaços de interação do agente. Por fim, a letra **M** corresponde ao conjunto de mecanismos existentes que servem para atualizar o estado do agente enquanto a simulação é executada (MACAL; NORTH; SAMUELSON, 2013).

Muitas vezes, os agentes possuem propriedades adicionais que podem ser consideradas ou não no modelo. Metas explícitas podem estar presentes em um agente, orientando o seu comportamento no modelo. Um agente tem a capacidade de aprender. Ele também pode ter a capacidade de se adaptar com base em suas experiências. A nível individual, essa aprendizagem e adaptação são modeladas através do comportamento do agente. Isso exige que o agente tenha uma memória como uma característica a ser dinamicamente atualizada. A nível de população, a adaptação pode ser modelada através de um conjunto de mudanças que resultam dos comportamentos de cada agente (MACAL; NORTH, 2014; SENA, 2017).

Fica claro, portanto, que são essenciais, em modelos de SBA, a modelagem do agente e de seus comportamentos, a modelagem dos comportamentos que afetam as próprias ações do agente, e a modelagem dos comportamentos que interferem no ambiente onde os mesmos se encontram.

A perspectiva do agente é a característica mais relevante e distintiva da SBA. Esse fato é discrepante com a ênfase dada em modelos de SED. A ênfase dos modelos de SED incide sobre processos ou atividades (MACAL, 2016). Por outro lado, um modelo de SBA é construído a partir da modelagem do agente.

Nos últimos 10 anos, houve divergências entre cientistas sobre a utilização da abordagem de simulação SBA na área de PO. Essas divergências ficam claras quando se expõe os títulos de dois artigos científicos: Siebers *et alii* (2010) e Brailsford (2014b). O título do primeiro é “*Discrete-event simulation is dead, long live agent-based simulation!*”; numa tradução livre seria “*A Simulação a Eventos Discretos está morta, vida longa à Simulação Baseada em Agentes!*”. Em resposta a esse artigo desenvolvido pelos pesquisadores P. O. Siebers, C. M. Macal, J. Garnett, D. Buxton e M. Pidd, S. Brailsford publica o artigo cujo título é “*Discrete-*

event simulation is alive and kicking!”; numa tradução livre seria “A Simulação de Eventos Discretos está viva e mandando ver!”.

Siebers *et alii* (2010) argumentam que a abordagem SBA, por ser capaz de modelar e simular ações individuais, pode auxiliar no melhor entendimento sobre os sistemas reais. A abordagem SBA possui maior aderência do que a abordagem SED ou SD quando se almeja capturar os efeitos produzidos por interações individuais. Os mesmos autores salientam que os pesquisadores que atuam na PO tendem a utilizar as abordagens de modelagem e simulação que estão mais familiarizados. Esse é um dos motivos que levam os pesquisadores a não adotar a abordagem SBA na modelagem e simulação de sistemas reais.

Ainda conforme Siebers *et alii* (2010), a abordagem SED é adequada em situações em que há a necessidade de simular sistemas de filas, pois em tais sistemas existem processos bem definidos e incertezas que são representadas nos modelos de SED por distribuições estocásticas. Essas características são encontradas em uma gama de sistemas de manufatura e serviços. Contudo, novos problemas estão exigindo dos pesquisadores que atuam na PO a modelagem e simulação de indivíduos que possuem ações e comportamentos diferentes nos sistemas reais. No Quadro 3.1 estão as principais distinções entre os modelos de SED e os modelos de SBA feitas por Siebers *et alii* (2010).

Quadro 3.1 - Características dos modelos de SED e de SBA

<i>Modelos de Simulação a Eventos Discretos</i>	<i>Modelos de Simulação Baseada em Agentes</i>
Orientado ao processo (abordagem de modelagem <i>top-down</i>); foco na modelagem do sistema em detalhe, não na modelagem das entidades	Com base no indivíduo (abordagem de modelagem <i>bottom-up</i>); foca na modelagem das entidades e das interações entre elas
Controle por meio de processos (controle centralizado)	Cada agente possui seu próprio controle de processo (controle descentralizado)
Entidades passivas; a inteligência, como a tomada de decisão, é modelada como parte do sistema	Entidades ativas; a inteligência é representada individualmente dentro de cada entidade
As filas são um elementos-chaves	Não apresenta nenhum conceito de filas
O fluxo de entidades ocorre através de um sistema; o comportamento macro é modelado	Não há nenhum conceito de fluxos; o comportamento macro não é modelado, pois ele emerge das micro decisões dos agentes individuais
As distribuições de entrada são muitas vezes baseadas na coleta e medição de dados	As distribuições de entrada são muitas vezes baseadas em teorias ou dados subjetivos

Fonte: Siebers *et alii* (2010)

Por fim, Siebers *et alii* (2010) ressaltam que a abordagem SBA não substitui as abordagens tradicionalmente utilizados na PO. Os autores também alertam que a abordagem SBA não é capaz de resolver todos os problemas e que também não é melhor do que as abordagens já consagradas na PO. Mas, é importante reconhecer a existência e a oportunidade em utilizar a abordagem SBA na resolução de problemas que acometem os sistemas reais.

Brailsford (2014b) reconhece que há um crescente interesse dos pesquisadores de PO sobre a SBA. Entretanto, a autora afirma que muitos dos benefícios da abordagem SBA citados no artigo de Siebers *et alii* (2010) podem ser alcançados por meio da abordagem SED. Ela ainda afirma que as abordagens SED e SBA possuem várias características em comum. Por exemplo, tanto as entidades de um modelo de SED quanto os agentes de um modelo de SBA possuem atributos que definem o seu comportamento no sistema retratado e as possíveis interações. Apesar das similaridades, a autora ressalta que as abordagens SED e SBA não são iguais e argumenta que qualquer modelo de SED pode ser replicado através da abordagem SBA, porém o mesmo pode não ocorrer quando se tenta reproduzir um modelo de SBA utilizando a abordagem SED. Sobre essa última afirmação, Brailsford (2014b) salienta que enquanto não houver um modelo de SBA que possa ser incontestavelmente impossível de ser replicado por meio da abordagem SED, ou o inverso, as discussões sobre o tema permanecerão.

Em seu artigo, Brailsford (2014b) relata que existe uma quantidade considerável de ferramentas criadas para a construção de modelos de SED e que elas permitem que o especialista desenvolva os seus modelos sem a necessidade de utilizar a linguagem de programação. Por outro lado, para se desenvolver um modelo de SBA, essa habilidade de programação é exigida do especialista. Conforme a sua experiência na área, a autora acrescenta que a quantidade de estudantes com essa habilidade de escrita computacional vem diminuindo.

Por fim, assim com Siebers *et alii* (2010), Brailsford (2014b) também considera que a familiaridade influencia na escolha da abordagem de modelagem. Além da familiaridade, outros elementos podem interferir na escolha da abordagem de modelagem como a disponibilidade do software e os recursos disponíveis para o projeto de simulação.

As abordagens SED e SBA possuem características divergentes. Expresso no Quadro 3.1, os modelos desenvolvidos por meio da abordagem SED possuem abordagem de modelagem *top-down*; o foco está na modelagem do sistema e seus processos. Já os modelos de SBA possuem abordagem de modelagem *bottom-up*; o foco está na modelagem das entidades e das

interações entre elas. Apesar dessas divergências, as duas abordagens de simulação se completam. Dessa união surge a Simulação Híbrida (SH).

3.2. Complexidade do Modelo de Simulação

Um projeto de simulação aborda problemas que envolve os sistemas reais. Sistema é um termo que comumente se observa em documentos científicos que reportam estudos desenvolvidos em diferentes áreas como física, matemática, química, biologia, letras, filosofia, engenharia, entre outras. Independente do campo científico, um “todo” é descrito, analisado e interpretado a partir dele mesmo e/ou de suas partes. Inicialmente, para compreender o que é complexo ou simples em um modelo computacional, é necessário definir conceitualmente o termo sistema.

O conceito sistema é aplicável no nível micro e no macro, e ele próprio define as fronteiras de análises. Por exemplo, em um contexto competitivo pode-se efetuar um conjunto de análises quanto aos elementos externos que potencializam a produtividade de uma organização. Nesse caso, os possíveis elementos a serem estudados estão na esfera macroscópica em que governo, mercado e tecnologia, *verbi gratia*, afetam diretamente a expansão ou contração dessa organização. Em suma, o conjunto de análises a ser efetuado coloca, invariavelmente, a organização dentro de um sistema composto por outros elementos que a afeta de modo positivo ou negativo. Nesse mesmo contexto competitivo, pode-se observar essa organização do ponto de vista micro ao analisar os elementos produtivos que geram maior ou menor competitividade.

Aplica-se o conceito sistema a qualquer "todo" no qual exista "componentes" em interação (BERTALANFFY, 1968). A união de componentes "isolados", as relações entre os componentes e a estrutura de camadas são características fundamentais de um sistema (FORREST, 2018). Sistema, do ponto de vista da Teoria Geral dos Sistemas, é um conjunto de componentes conectados que, dependendo do nível de análise, cada componente pode ser tratado como um sistema composto por outros componentes conectados, e assim por diante, dando a ideia de camadas.

O termo sistema pode também ser interpretado como uma união de objetos que possuem, de modo regular, uma interação ou interdependência (GORDON, 1969). Em outras palavras, uma composição de componentes interdependentes que se interagem é um sistema (POPOVICS; MONOSTORI, 2016). Assim, no campo científico da Simulação, sistema significa uma união de elementos correlatos que se relacionam regularmente.

Nessa tese, o termo sistema significa concentração de elementos interdependentes que se interagem de maneira regular e que pode ser decomposto em outros sistemas menores. Seguindo esse raciocínio, o conceito de complexidade ou simplicidade está vinculada ao conceito de sistema.

Em termos de tamanho, os sistemas desenvolvidos pela humanidade estão crescendo continuamente e, intuitivamente, esse fato aumenta a complexidade de tais sistemas. Os exemplos são muitos, como é o caso de empresas de manufatura, logística, saúde e finanças que buscam atender às necessidades da demanda crescente por meio de sistemas de engenharia que buscam combinar o mundo digital com o real. Portanto, não é inesperado o aumento do tamanho e da complexidade dos sistemas (SARJOUGHIAN, 2017).

O aumento da complexidade de um dado sistema ocorre, de modo geral, quando há um crescimento no número de componentes e na quantidade de conexões entre eles (POPOVICS; MONOSTORI, 2016). Isso significa que o aumento do tamanho e da complexidade dos sistemas reais gera o mesmo efeito nos modelos conceituais e computacionais, o que acaba se tornando um dilema em um projeto de simulação: se o especialista cria um modelo computacional detalhado, o tempo do projeto se estende e o modelo pode ser validado ou não; se o especialista cria um modelo computacional menos detalhado, o tempo do projeto é otimizado e o modelo pode ser validado ou não. Portanto, na prática, o especialista em simulação busca representar, minimamente e de modo fidedigno, um sistema real por meio de um modelo computacional. O desafio é: o quão é esse mínimo que o modelo precisa ter para representar um sistema real de forma confiável?

Conforme Brooks e Tobias (1996), espera-se que um modelo computacional produza resultados que descrevam o comportamento da variável de interesse e que esses resultados sejam precisos. Além disso, espera-se também que haja facilidade em compreender o modelo computacional e os resultados gerados por ele. O alcance desses resultados depende do nível de detalhe adotado para descrever o sistema em estudo.

Um modelo cumpre a função de reproduzir, de modo diverso, um objeto, um sistema ou uma ideia (SHANNON, 1975). Para Bertrand e Fransoo (2002), um modelo consiste sempre em uma abstração da realidade que não inclui todos os elementos dessa realidade. Robinson (2013) estabelece que um modelo é uma representação simplificada da realidade construído para uma finalidade específica; dessa maneira, não se pode afirmar que esse modelo é um modelo geral da realidade.

O conjunto de relações lógicas e causais que ocorrem nos sistemas reais é o cerne de um modelo de simulação (SARGENT, 2014). Baseado em Little (1970), Shannon (1975, p. 22, grifo do autor) define que um bom modelo de simulação precisa conter os seguintes atributos:

- ***Simples*** de ser entendido pelo usuário;
- ***Meta*** ou propósito direcionado;
- ***Robusto***, na medida em que respostas absurdas não são dadas;
- ***Fácil de ser controlado*** ou manipulado pelo usuário, ou seja, deve ser fácil de se comunicar;
- ***Completo*** acerca de questões importantes;
- ***Adaptativo***, com um procedimento fácil para efetuar uma modificação ou atualização do modelo;
- ***Evolutivo***, na medida em que o modelo deve iniciar de forma simples e se tornar mais complexos, em junção com o usuário.

Etimologicamente, a palavra complexidade vem do termo latim *complexus* que significa entrelaçado. Para existir complexidade é necessário que haja componentes e conexões entre eles. Intuitivamente, o sistema aparenta ser mais complexo quanto maior for o número de componentes e de conexões entre eles (BERTUGLIA; VAIO, 2005). Para Zeigler *et alii* (2000), a complexidade de um modelo depende do número de componentes e de variáveis presentes no mesmo.

Segundo Yücesan e Schruben (1998), a complexidade de modelos de simulação é uma medida que reflete os requisitos necessários para construção de um modelo. Essa complexidade está geralmente relacionada com as propriedades estruturais do modelo.

Assim, as propriedades estruturais de um modelo de simulação envolvem componentes, conexões, estruturas lógicas, fluxo de informações, linhas de código, requisitos de hardware e software. Nessa tese, a complexidade de um modelo computacional é entendida como um reflexo da quantidade de componentes, do número de conexões entre os componentes e a essência dessas conexões.

Sarjoughian (2017) afirma que a complexidade é frequentemente interpretada como uma forma de hierarquia em que um sistema complexo é composto por subsistemas inter-relacionados que, por sua vez, possuem seus próprios subsistemas. Esse pensamento sobre um

dado sistema vai até chegar no nível mais baixo no qual existem componentes elementares. Ademais, o mesmo autor afirma que o conceito de heterogeneidade possui uma relação direta com o nível de complexidade dos sistemas.

O que determina a simplicidade ou a complexidade de uma estrutura é o modo crítico como ela é descrita (SIMON, 1962). Na área de simulação, a complexidade está associada ao nível de detalhe dos modelos computacionais. O nível de detalhe de um modelo computacional refere-se mais ao sistema que esse modelo representa do que o próprio modelo. Em outras palavras, dependendo do escopo adotado pelo projeto de simulação, a adoção de um nível de detalhe está relacionada à avaliação de inclusão de elementos e de suas conexões que são observáveis no sistema em estudo (BROOKS; TOBIAS, 1996). A escolha do nível de detalhe influencia na complexidade do modelo. A Figura 3.1, baseada na pesquisa de Robinson (2008), expressa a relação entre o nível de detalhamento e a precisão do modelo computacional.

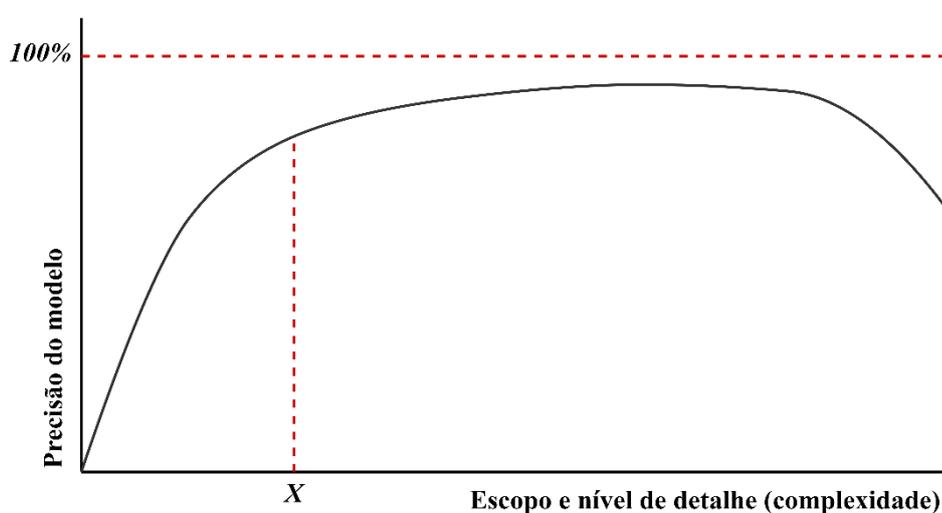


Figura 3.1 - Complexidade e precisão do modelo de simulação

Fonte: Adaptado de Robinson (2008)

Robinson (2008) argumenta que um modelo 100% preciso não é possível de ser alcançado devido a impossibilidade de capturar todas as variáveis presentes em um sistema real. Além disso, o aumento do nível de detalhamento do modelo computacional pode torná-lo menos preciso. Por fim, o *X* da questão é encontrar um *ponto ideal* em que a captura dos principais elementos do sistema real seja satisfatória.

É exatamente esse *ponto ideal* que gera discussões na área científica da simulação quanto a utilização ou não da combinação das abordagens de simulação. Este é o principal motivo da existência dessa tese cujo objetivo central gira em torno da discussão sobre o nível de

detalhamento dos modelos computacionais. Como afirmado anteriormente, essa discussão será feita a partir de análises sobre a complexidade dos modelos computacionais.

A avaliação da complexidade de um sistema é um ato que permite que as escolhas de modelagem sejam justificadas ou alteradas em função da própria complexidade encontrada (MAYAKA; JUNG, 2018). Métricas quantitativas de complexidade foram desenvolvidas a partir do esforço de muitos engenheiros de software. Essas métricas podem focar no código, na modelagem, na manutenção, na documentação, na verificação ou nas especificações necessárias para o desenvolvimento de um sistema (YÜCESAN; SCHRUBEN, 1998). Por exemplo, Hasić e Vanthienen (2019) aplicam diferentes métricas de complexidade para mensurar a complexidade de modelos conceituais.

Polančič e Cegnar (2017), por meio de uma revisão sistemática da literatura, mostram as principais métricas de complexidade utilizadas no campo científico. Segundo esses autores, a métrica de complexidade Complexidade Ciclométrica (CC) de McCabe (1976) é a mais utilizada. Em segundo e terceiro lugar vêm, respectivamente, as métricas Linha de Código (LC) (KALB, 1990) e Tamanho Funcional Cognitivo (TFC) (SHAO; WANG, 2003).

A CC exibe a complexidade de um sistema com base no número de caminhos de controle existentes em um módulo de programa. Com base na teoria dos grafos, a equação de CC usa a definição de número ciclométrico que consiste no número de caminhos linearmente independentes dentro de um conjunto de códigos (MCCABE, 1976).

A LC é uma medida de complexidade baseada na análise de código por meio da contagem das linhas de código de um programa. Essa medida é simples e, com base no tamanho do programa, ela pode servir como indicador para análise de erros, confiabilidade e facilidade de manutenção (KALB, 1990).

O TFC é uma medida de complexidade que atribui um peso cognitivo sobre cada estrutura básica de controle, como por exemplo: número de declarações, lógica *if-then* ou *if-then-else*, número de funções, lógica *for-do*, lógica *repeat-until*, lógica *while-do*, e outros. Os pesos para cada estrutura lógica foram definidos por Shao e Wang (2003) com base em estudos empíricos.

Nessa tese a métrica de complexidade que será utilizada para atribuir pesos às conexões entre os componentes existentes em um modelo computacional será a CC de McCabe (1976). Essa escolha está alinhada com o fato de que essa métrica de complexidade é a mais utilizada no meio científico.

4. A METODOLOGIA DE PESQUISA

A Pesquisa Operacional (PO) é uma das áreas de estudo da Engenharia da Produção. Situada dentro da PO, a Modelagem, Simulação e Otimização possui um campo de estudo baseado na construção de modelos que têm por finalidade representar sistemas reais. Esses modelos são de caráter quantitativo e procuram absorver as principais características presentes no sistema real (BERTRAND; FRANSOO, 2002; CHWIF; MEDINA, 2015). Nesse sentido, essa pesquisa fará uso de modelos computacionais para construir – por meio de diferentes abordagens de simulação – um sistema de manufatura hipotético composto por três postos de trabalho. Posteriormente, serão feitas análises acerca da complexidade desses modelos computacionais.

Do ponto de vista da natureza científica, essa pesquisa se enquadra como pesquisa aplicada, pois ela tem como foco a geração de conhecimentos para aplicação prática de um problema específico envolvendo a área de simulação. Acerca da classificação dos objetivos científicos, essa pesquisa é explicativa devido à preocupação em analisar e interpretar os fatores intrínsecos à complexidade de modelos computacionais. Quanto aos procedimentos técnicos, essa pesquisa é de caráter experimental, porque os modelos computacionais, na condição de objetos de estudo, podem ser manipulados, testados e observados (PRODANOV; FREITAS, 2013).

A presente tese segue uma abordagem de pesquisa quantitativa, quatro modelos computacionais serão criados com o objetivo de representar de maneira adequada as principais características de uma linha de produção do ramo de manufatura. O foco está em analisar o grau de complexidade dos modelos considerando (i) a abordagem de simulação adotada e (ii) o nível de detalhamento dos modelos. Para aplicar a abordagem de pesquisa quantitativa, o método científico adotado nessa pesquisa é a Modelagem e Simulação.

4.1. Modelagem e Simulação

Sendo uma abstração da realidade, modelos quantitativos são criados através do método científico Modelagem e Simulação. Isso permite ao pesquisador a manipulação das variáveis utilizando ou não ferramentas computacionais (MARTINS, 2012).

Há dois tipos específicos de pesquisa baseada em modelos quantitativos na área de Modelagem e Simulação. O primeiro tipo recebe o nome de pesquisa axiomática quantitativa. O pesquisador que trabalha com esse tipo de pesquisa atenta-se em obter soluções que

auxiliam no esclarecimento do problema descrito no modelo. Além disso, métodos matemáticos, estatísticos e computacionais são usados de modo intenso no contexto da pesquisa axiomática quantitativa. O segundo tipo de pesquisa é denominado pesquisa empírica quantitativa. Nesse tipo de pesquisa, o pesquisador preocupa-se em garantir a adesão do modelo elaborado com as observações e ações na realidade. Vale acrescentar que, na pesquisa empírica quantitativa, os modelos são criados para se adequarem às relações causais do problema real (BERTRAND; FRANSOO, 2002; MORABITO NETO; PUREZA, 2012).

Os dois tipos de pesquisa supracitados ainda são classificados em duas categorias: descritiva e normativa. Assim, temos a pesquisa axiomática descritiva, pesquisa axiomática normativa, pesquisa empírica descritiva e pesquisa empírica normativa. Na pesquisa axiomática descritiva, o foco do pesquisador está na análise do modelo quantitativo. Na pesquisa axiomática normativa, o pesquisador concentra-se em desenvolver um modelo de solução do processo. Na pesquisa empírica descritiva, o pesquisador foca no desenvolvimento de um modelo que tenha a capacidade de descrever adequadamente as relações causais existentes na realidade, levando a uma melhor compreensão dos processos reais. Na pesquisa empírica normativa há a preocupação em elaborar políticas, estratégias e ações para melhorar a atual situação. Além disso, nesse tipo de pesquisa o pesquisador cria modelos que determinam a tomada de decisão para um certo tipo de problema (BERTRAND; FRANSOO, 2002; MORABITO NETO; PUREZA, 2012).

Nesse momento faz-se necessária a definição de qual tipo específico de pesquisa baseada em modelos quantitativos que essa tese irá adotar. Por se tratar de uma pesquisa que envolve a mensuração da complexidade de modelos computacionais por meio de métodos matemáticos e estatísticos, conseqüentemente, dentre os tipos específicos de pesquisa baseada em modelos quantitativos, a presente tese caracteriza-se como pesquisa axiomática quantitativa.

Na PO existem diversas técnicas voltadas para o desenvolvimento de modelos conceituais e computacionais. Uma dessas técnicas é aquela proposta por Mitroff *et alii* (1974), composta por quatro elementos que orientam o pesquisador na criação de modelos quantitativos na área de PO. Outra técnica difundida na literatura científica acerca do processo de construção de um projeto de simulação é a de Arenales *et alii* (2007). Contudo, essa tese fará uso da técnica proposta pelo pesquisador Sargent (2013). A justificativa quanto a escolha dessa técnica, retratada pela Figura 4.1, está na facilidade em relatar em que ponto o *método* desenvolvido nessa tese se encaixa em um projeto de simulação.

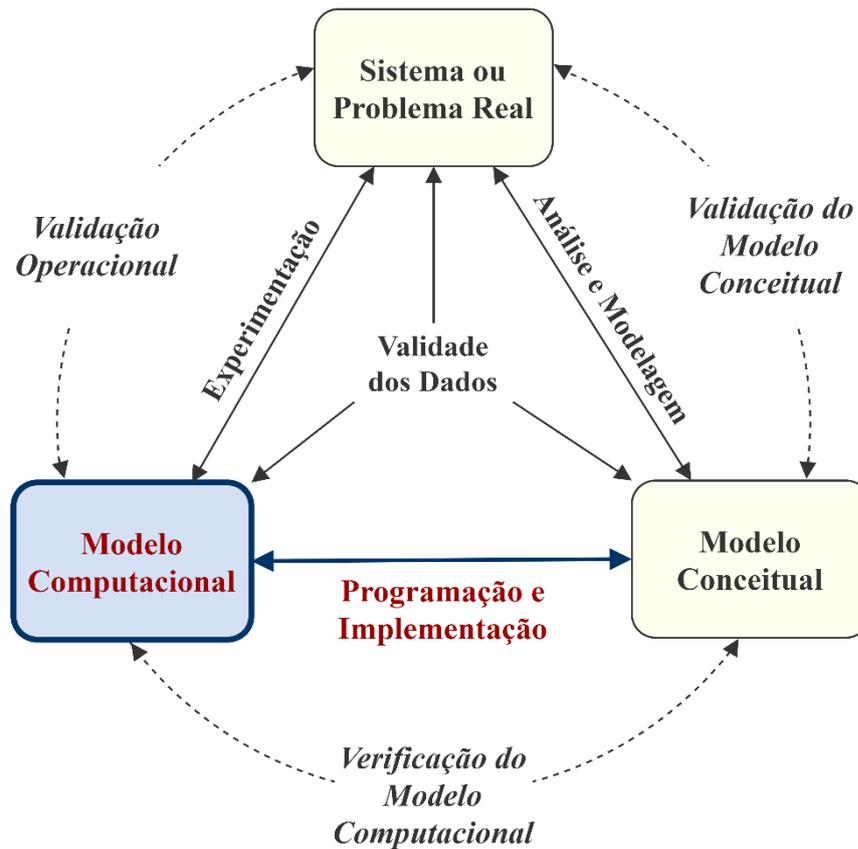


Figura 4.1 - Técnica de desenvolvimento de um modelo de simulação
 Fonte: Adaptado de Sargent (2013)

Na Figura 4.1, em destaque estão os termos (i) *Programação e Implementação* e (ii) *Modelo Computacional*. No termo (i) está a transformação do Modelo Conceitual em Modelo Computacional. Essa transformação ocorre a partir da aplicação de uma das abordagens de simulação. No caso dessa tese serão duas abordagens que serão aplicadas – SED e SH (SBA+SED). Com Modelo Computacional desenvolvido, inicia-se o processo de mensuração da complexidade do modelo por meio da aplicação do Método iDAV. Esse processo será explicado com maior detalhe na próxima seção.

4.2. Método iDAV

O método desenvolvido por esse trabalho de doutorado visa atender aos objetivos específicos traçados na Seção 1.1. Busca-se, por meio deste método, mensurar a complexidade estrutural dos modelos de simulação e definir parâmetros metodológicos que irão auxiliar no processo de mensuração desses modelos. O nome do método é iDAV e esse nome é o acrônimo dos termos *identificação*, *Definição*, *Análise* e *Verificação* (BARBIERI, PINHO; MONTEVECHI, 2023). Esses termos representam a fases do processo metodológico de mensuração da complexidade estrutural dos modelos computacionais.

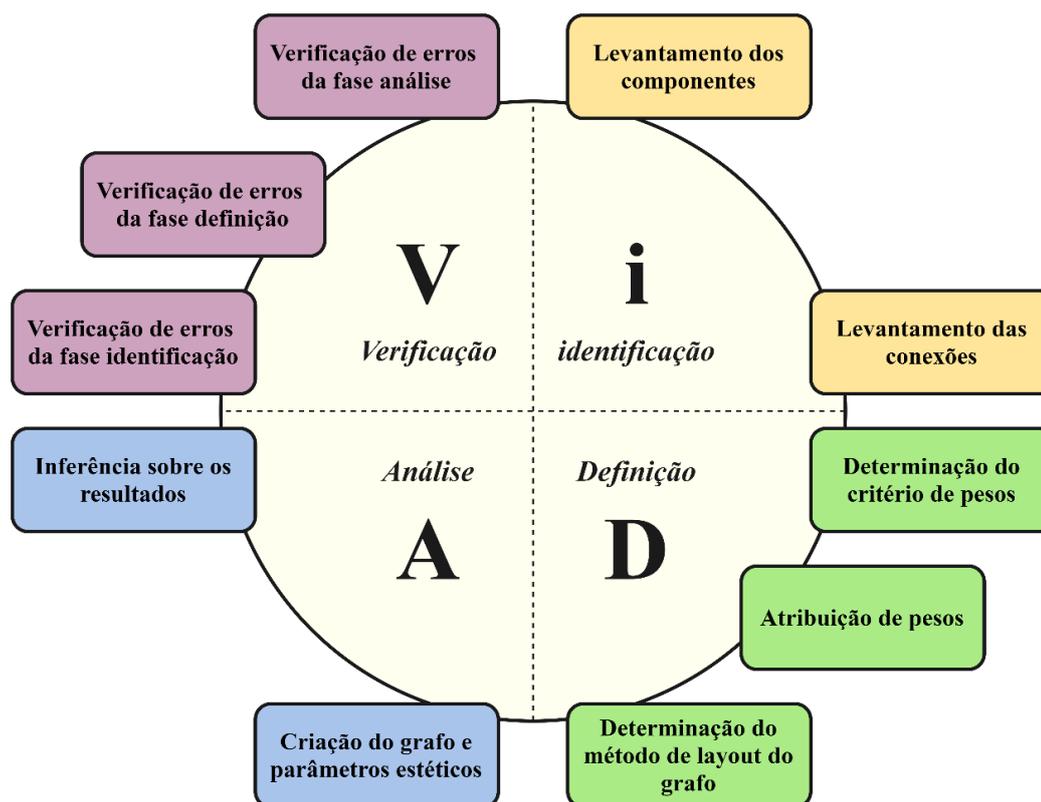


Figura 4.2 - Método iDAV

Fonte: Elaborado pelo próprio autor

Conforme a Figura 4.2, o método iDAV possui característica cíclica. Ele composto por quatro fases e o processo de mensuração da complexidade de modelos computacionais inicia-se na fase de *identificação* – esse termo começa com letra minúscula para indicar o início do processo. Na sequência, são colocadas em prática as fases *Definição*, *Análise* e *Verificação*.

A fase *identificação* é composta por duas ações que são (i) o **Levantamento dos componentes** e (ii) o **Levantamento das conexões**. Nessa fase, os componentes existentes no modelo computacional são identificados e registrados. O mesmo é feito com as conexões desses componentes, porém é fundamental documentar a relação entre cada componente. O resultado dessa fase se materializa em uma matriz que pode ser construída por meio de planilhas eletrônicas.

Na fase *Definição* existem três etapas que são (i) **Determinação do critério de pesos**, (ii) **Atribuição de pesos** e (iii) **Determinação do método de layout do grafo**. Na etapa (i) é determinado o critério de pesos a ser adotado em cada conexão existente entre os componentes. Em seguida, os pesos são atribuídos da etapa (ii) dentro da matriz. Na etapa (iii) é definido o método de layout do grafo. Nesse ponto é importante ressaltar que o *método* iDAV contém conceitos oriundos da Teoria dos Grafos. Assim, para ajudar na definição do

layout do grafo, recomenda-se os trabalhos de Godehardt (1988), Di Battista *et alii* (1994), Herman, Melançon e Marshall (2000), Kaufmann e Wagner (2001), Koren (2005), Wallis (2007), Jacomy *et alii* (2014) e Tamassia (2014).

Na fase *Análise* estão as etapas (i) **Criação do grafo e parâmetros estéticos** e (ii) **Inferência sobre os resultados**. Existem ferramentas que auxiliam na criação e na definição de parâmetros estéticos dos grafos. Uma dessas ferramentas é o software Gephi® – software de código aberto utilizado para análise e visualização de redes. O Gephi® foi escolhido por ser um software livre e por possuir mecanismos que facilitam a comunicação com planilhas eletrônicas. Com o grafo criado, o processo de inferência sobre os resultados se inicia. Essa inferência é feita com conceitos e métodos estatísticos.

A fase de *Verificação* é a última do método iDAV. Essa fase é composta por três atividades que são (i) **Verificação de erros da fase identificação**, (ii) **Verificação de erros da fase definição** e (iii) **Verificação de erros da fase análise**. Tendo em mente o aprimoramento contínuo do processo de mensuração da complexidade dos modelos computacionais, a fase de *Verificação* foi dividida em três etapas de modo a determinar o foco da ação de melhoria. Assim, nessa fase, o especialista verifica a existência de erros que podem ter ocorrido nas fases anteriores.

4.3. Modelos Computacionais

O objetivo geral dessa pesquisa é mensurar a complexidade de modelos computacionais discretos e híbridos, portanto, para atingir esse objetivo, essa pesquisa irá trabalhar com um sistema hipotético no ramo de manufatura no qual há dois níveis de detalhamento. Os motivos que reforçam a escolha de trabalhar com modelos hipotéticos são os seguintes:

- **Tipo do sistema.** São diversos os tipos de sistemas que rodeiam a humanidade, portanto é fundamental delimitar o tipo de sistema a ser estudado para que assim seja possível analisá-lo. Escolher um sistema no qual existe conhecimento prévio sobre os fluxos de informações e materiais, auxilia na construção de um modelo mais próximo da realidade. Ademais, nem sempre é possível ter acesso a todas as variáveis que influenciam o sistema, pois existem questões que limitam a atuação do pesquisador dentro do ambiente organizacional. Portanto, a liberdade em escolher o tipo de sistema a ser abordado, no caso dessa pesquisa, facilita a análise mais acurada o objeto em estudo.

- **Essência do sistema.** Modelos hipotéticos dão a possibilidade de controlar a quantidade de elementos e as relações entre ele. Essa liberdade é restringida quando se estuda um sistema real no qual, geralmente, possui particularidades. Um sistema real, por exemplo, em um primeiro momento, pode aparentar ser simples, porém o fluxo de informações pode ter características opostas. Assim, trabalhar com modelos hipotéticos possibilita um maior controle sobre as variáveis que afetam o sistema. Por fim, o foco dessa pesquisa está sobre o modelo computacional e não sobre o sistema real.

Nessa pesquisa será adotado um sistema de manufatura com layout em linha. A escolha desse tipo de sistema está relacionada com o conhecimento que o pesquisador possui sobre as particularidades gerais desse sistema. Além disso, pode-se destacar que um sistema de manufatura em linha é de natureza simples e ele permite explorar os níveis de decisão humana dentro das limitações que o próprio sistema impõe.

Todos os modelos computacionais, discretos e híbridos, foram construídos a partir do modelo conceitual expresso pela Figura 4.3. O IDEF-SIM (*Integrated Definition Methods – Simulation*), técnica de modelagem desenvolvida por Leal (2008), foi a ferramenta utilizada para o desenvolvimento do modelo conceitual. Essa técnica é adotada nessa tese, pois, conforme Leal (2008), o IDEF-SIM auxilia na redução de tempo durante a fase de modelagem computacional.

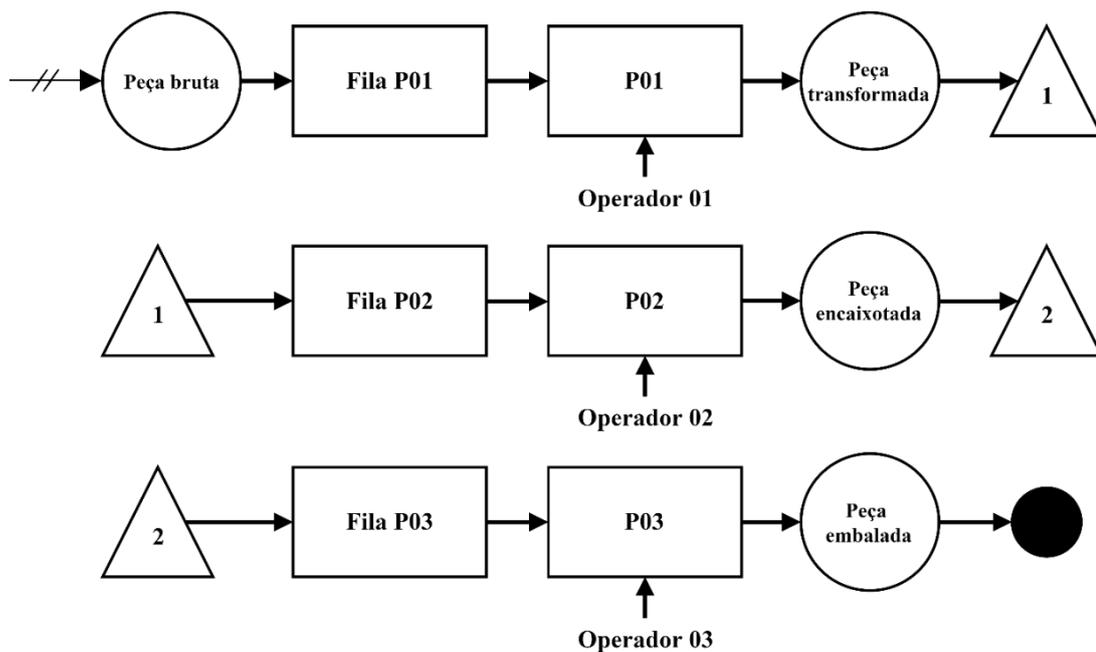


Figura 4.3 - Modelo conceitual do sistema de manufatura
Fonte: Elaborado pelo próprio autor

O modelo conceitual do sistema hipotético mostra três processos que comumente ocorre em sistemas de manufatura. Esses processos estão em linha e cada um deles possui uma fila. Há, em cada processo, um operador. O Operador 1 recebe a peça bruta e inicia o processo de transformação dessa peça. O Operador 2 recebe a peça transformada e a coloca dentro de uma caixa. O Operador 3 recebe a caixa e a embala para expedição.

As distribuições de probabilidade das operações P01, P02 e P03 são, respectivamente, Normal (61,33; 8,29), Normal (70,14; 11,18) e Normal (50,97; 5,18). Cada posto de trabalho possui a capacidade de fila igual a 1. No final da linha há uma variável que contabiliza quantas peças foram encaixotadas e embaladas.

Com base na Figura 4.3, foram construídos os modelos computacionais D0N3, H0N3, D1N3 e H1N3 por meio do AnyLogic®. O que diferencia um modelo de outro são dois fatores: a abordagem de simulação utilizada e o grau de autonomia das entidades/agentes. Portanto, quando se lê D0N3, a letra **D** significa que o modelo computacional foi construído exclusivamente com a abordagem SED, o número **0** (zero) significa que os operadores não possuem características autônomas no modelo, a letra **N** significa que o modelo não possui elementos relacionados à simulação 3D, e o número **3** (três) indica que o modelo possui três postos de trabalho.

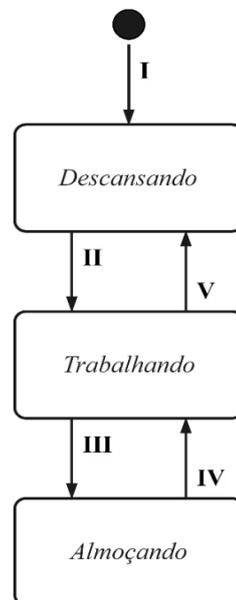


Figura 4.4 - Diagrama de estado dos operadores dos modelos D0N3 e H0N3

Fonte: Elaborado pelo próprio autor

Conforme a Figura 4.4, os operadores possuem três estados: “*Descansando*”, “*Trabalhando*” e “*Almoçando*”. A simulação começa (**I**) com os operadores no estado “*Descansando*”. Às 8:00 (**II**), os operadores iniciam o trabalho e passam do estado “*Descansando*” para o estado

“Trabalhando”. Às 12:00 (III), os operadores vão almoçar e retornam do estado “Almoçando” para o estado “Trabalhando” às 13:00 (IV). Às 17:00 (V), o trabalhador passa do estado “Trabalhando” para o estado “Descansando”. Esse fluxo se repete 5 (cinco) vezes durante uma semana de trabalho.

O diagrama de estado, retratado pela Figura 4.4, mostra o fluxo lógico no qual se baseia o comportamento dos operadores tanto no modelo DON3 quanto no modelo HON3. Na abordagem SED, por não se fazer uso do diagrama de estado (INGALLS, 2008), o comportamento de uma entidade é norteado por eventos, calendários e variáveis. Assim, cada operador do modelo DON3 possui um calendário que define os horários de entrada e saída dos operadores. Além disso, existem três variáveis que estão vinculadas a cada calendário. Essas variáveis desempenham a função lógica de expor qual o atual estado da entidade. As mudanças de um estado para outro ocorrem por meio da linguagem de programação JAVA, utilizando a lógica *if – else if – else*.

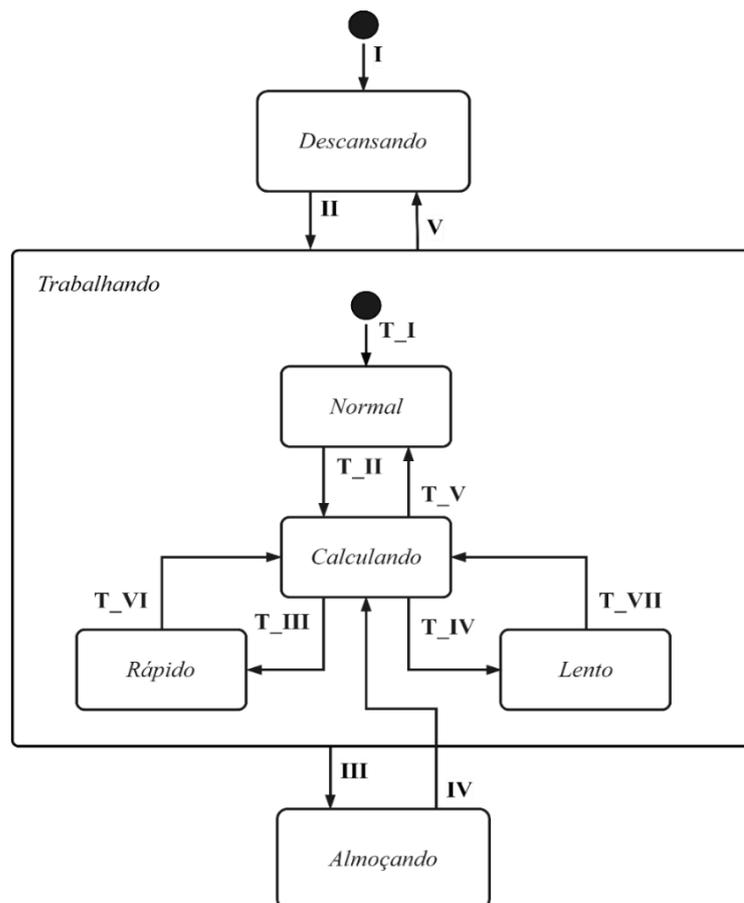


Figura 4.5 - Diagrama de estado dos operadores dos modelos D1N3 e H1N3

Fonte: Elaborado pelo próprio autor

No modelo HON3, a utilização da abordagem SBA gera a possibilidade de desenvolver o diagrama de estado de cada operador. Portanto, tendo como base a Figura 4.4, os estados dos

agentes e as suas transições (as setas) foram criados no ambiente virtual. Há, para cada agente, um calendário vinculado à uma variável que auxilia na mudança de um estado para outro. Assim como no modelo D0N3, a linguagem JAVA também é empregada, porém a lógica *if-else if-else* não é aplicada devido ao uso do diagrama de estado e suas funções.

Os operadores dos modelos computacionais D1N3 e H1N3 também possuem os estados “*Descansando*”, “*Trabalhando*” e “*Almoçando*”. Como dito anteriormente, o que diferencia os modelos D0N3 e H0N3 dos modelos D1N3 e H1N3 é o grau de autonomia das entidades/agentes. Então, conforme a Figura 4.5, houve o acréscimo de quatro estados dentro de “*Trabalhando*”. Esses estados são: “*Normal*”, “*Lento*”, “*Rápido*” e “*Calculando*”. Os três primeiros estados reproduzem o ritmo de trabalho dos operadores com base no Sistema Westinghouse (BARNES, 1977) que considera quatro parâmetros: Habilidade, Esforço, Consistência e Condições. Na Tabela 4.1, estão os valores de cada um desses parâmetros. O estado “*Calculando*” possui a função de ligação entre os demais estados por meio dos cálculos. *Verbi gratia*, o operador está no estado “*Lento*”, após uma hora ele verifica a sua eficiência e, com base nela, ele altera o seu ritmo de trabalho para a próxima hora de trabalho.

Tabela 4.1 - Estimativas de desempenho

HABILIDADE			ESFORÇO		
+0,15	A1	<i>Super-hábil</i>	+0,13	A1	<i>Excessivo</i>
+0,13	A2		+0,12	A2	
+0,11	B1	<i>Excelente</i>	+0,10	B1	<i>Excelente</i>
+0,08	B2		+0,08	B2	
+0,06	C1	<i>Bom</i>	+0,05	C1	<i>Bom</i>
+0,03	C2		+0,02	C2	
0,00	D	<i>Médio</i>	0,00	D	<i>Médio</i>
-0,05	E1	<i>Regular</i>	-0,04	E1	<i>Regular</i>
-0,10	E2		-0,08	E2	
-0,16	F1	<i>Fraço</i>	-0,12	F1	<i>Fraço</i>
-0,22	F2		-0,17	F2	
CONDIÇÕES			CONSISTÊNCIA		
+0,06	A	<i>Ideal</i>	+0,04	A	<i>Perfeita</i>
+0,04	B	<i>Excelente</i>	+0,03	B	<i>Excelente</i>
+0,02	C	<i>Boa</i>	+0,01	C	<i>Boa</i>
0,00	D	<i>Média</i>	0,00	D	<i>Média</i>
-0,03	E	<i>Regular</i>	-0,02	E	<i>Regular</i>
-0,07	F	<i>Fraca</i>	-0,04	F	<i>Fraca</i>

Fonte: Adaptado de Barnes (1977)

No modelo D1N3, o estado “*Calculando*” é o próprio bloco de código computacional. É a partir do resultado desse bloco que a entidade define o seu ritmo de trabalho no decorrer da simulação. É importante dizer que esse bloco de código utiliza a lógica *if – else if – else* dentro da lógica *if – else* e ele se encontra dentro de um dos calendários.

Cada operador do modelo D1N3 possui dois calendários, nove variáveis e vinte e sete parâmetros. Um dos dois calendários e três das nove variáveis já existiam no modelo D0N3. O outro calendário se encontra vinculado às seis variáveis que definem o estado da entidade, o que foi produzido na última hora, a meta que o operador deveria cumprir na última hora, a eficiência da entidade na última hora e o ritmo da entidade para a próxima hora. Esse calendário também está vinculado aos vinte e sete parâmetros que possuem o valor total de horas trabalhadas, o valor global de produção, a meta de produção para o período e os valores da Tabela 4.1 utilizados para calcular o ritmo de trabalho.

No modelo H1N3, o diagrama de estado foi construído para cada operador, segundo a Figura 4.5, e foram adicionados quatro variáveis e vinte e sete parâmetros. Essas variáveis expõem quanto foi produzido na última hora, a meta que deveria ser cumprida na última hora, a eficiência do agente na última hora e o ritmo que o agente adotará para a próxima hora. Os vinte e sete parâmetros estão relacionados com os valores da Tabela 4.1, o valor global de produção, o valor total das horas trabalhadas e meta de produção definida para o período.

Assim como nos modelos computacionais D0N3 e H0N3, os operadores dos modelos D1N3 e H1N3 cumprem os horários de entrada e saída, percorrendo os estados “*Descansando*”, “*Trabalhando*” e “*Almoçando*”. No estado “*Trabalhando*”, tanto no modelo D1N3 quanto no modelo H1N3, os operadores começam o trabalho no ritmo “*Normal*”. Após uma hora de trabalho, os operadores verificam quanto foi produzido e quanto deveria ser produzido. Essa verificação ocorre na passagem no estado “*Calculando*”. Dependendo do valor expresso pela variável “*Eficiência*”, razão de quanto foi produzido e quanto deveria ser produzido, os operadores decidem se devem aumentar o seu ritmo passando para o estado “*Rápido*”, se devem diminuir o seu ritmo adotando o estado “*Lento*” ou se devem manter o seu ritmo permanecendo no estado “*Normal*”.

Com os modelos computacionais D0N3, H0N3, D1N3 e H1N3 construídos, é necessário verificar, por meio de métodos estatísticos, se os resultados dos modelos são estatisticamente iguais para que seja possível efetuar comparações e análises de sensibilidade. Assim, de acordo com a Figura 4.6, foi feita uma comparação entre os modelos computacionais D0N3,

H0N3, D1N3 e H1N3, utilizando a variável que representa o total de peças encaixotadas por dia.

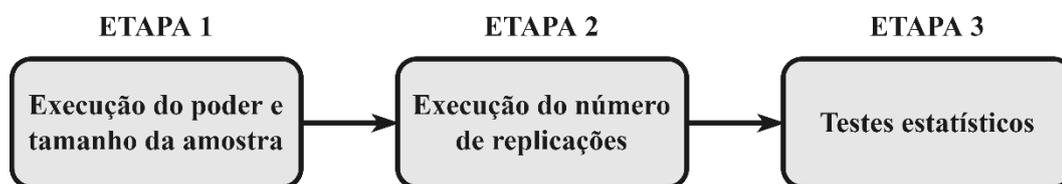


Figura 4.6 - Procedimento estatístico para a validação dos modelos computacionais

Fonte: Elaborado pelo próprio autor

Na Etapa 1, o teste estatístico Poder e Tamanho de Amostra para One-Way ANOVA foi executado para determinar o tamanho da amostra considerando a diferença mínima de uma peça embalada entre os quatro modelos. Considerando um valor de poder de 0,99, o tamanho da amostra foi de 1695 replicações. Na Etapa 2, 1695 replicações de cada modelo foram executadas.

Na Etapa 03, o Teste de Igualdade de Variâncias foi feito, para constatar se as variâncias dos modelos são diferentes, e o Teste *One-Way* ANOVA, para determinar se as médias dos modelos são diferentes. O nível de significância para esses testes foi estabelecido em 0,05. No Teste de Igualdade de Variâncias, considerando os intervalos de confiança de Bonferroni de 95%, deve-se rejeitar a hipótese nula de que todas as variâncias dos modelos são iguais, já que o *P-value* é igual à 0,000. Em razão disso, o método Games-Howell foi utilizado no Teste *One-Way* ANOVA. O resultado deste teste estatístico indica a hipótese nula não deve ser rejeitada, pois as médias dos modelos são estatisticamente iguais, como mostra o *P-value* cujo valor é 0,617.

Com o procedimento estatístico finalizado, o método iDAV pode ser aplicado para cada modelo computacional. Assim, cada modelo passará por processos de identificação dos componentes e conexões, de definição de critérios para criação dos grafos, de análise das características dos grafos, e de verificação de possíveis erros.

5. A APLICAÇÃO DO MÉTODO iDAV

5.1. identificação

O intuito dessa seção e das próximas é descrever o procedimento de mensuração da complexidade dos modelos computacionais. Para mensurar a complexidade dos modelos computacionais na fase de *identificação*, as seguintes etapas foram executadas: (i) **levantamento dos componentes** e o (ii) **levantamento das conexões**.

Na etapa de **levantamento dos componentes**, foram contabilizados 31 componentes do modelo D0N3, 48 componentes do modelo H0N3, 133 componentes do modelo D1N3 e 174 componentes do modelo H1N3. Entende-se por componente todo elemento inserido pelo especialista no ambiente de simulação.

Na etapa de **levantamento das conexões**, verificou-se as conexões existentes entre cada componente do modelo. Assim, foram contabilizadas 36 conexões no modelo D0N3, 78 conexões no modelo H0N3, 180 conexões no modelo D1N3 e 378 conexões no modelo H1N3. A Tabela 5.1 mostra um resumo dos dados coletados dos modelos computacionais.

Tabela 5.1 - Resultados do procedimento de contagem

Modelos computacionais	Nº de componentes	Nº de conexões		
		Sem linha de código	Com linha de código	Total
D0N3	31	22	14	36
H0N3	48	49	29	78
D1N3	133	22	158	180
H1N3	174	91	287	378

Fonte: Elaborado pelo próprio autor

As conexões existentes dentro de um modelo possuem naturezas distintas. Nota-se, a partir dos dados da Tabela 5.1, que existem conexões entre componentes nas quais não há a necessidade de escrever linhas de código. Por outro lado, quando não se consegue efetuar a ação *drag-and-drop* (arrastar e soltar), o especialista usa o recurso de programação computacional para conectar os componentes dentro do modelo. Os resultados encontrados na Tabela 5.1 serão analisados na seção 5.3 na qual será abordada a fase *Análise*.

5.2. Definição

Nessa seção, serão tratadas as etapas: (i) **determinação do critério de pesos**, (ii) **atribuição de pesos** e (iii) **determinação do método de layout do grafo**. Durante o procedimento de contagem das conexões existentes entre os componentes de um dos modelos computacionais, percebeu-se que a praticidade em utilizar a ação *drag-and-drop* em comparação à construção de linhas de códigos era maior. Todavia, essa praticidade é relativa e depende do nível de conhecimento e de experiência do especialista em simulação. Assim, atribuir peso às conexões se baseando no conhecimento e na experiência do especialista traz resultados tendenciosos. Com intuito de eliminar essa tendência, os pesos foram atribuídos com base na Equação 5.1.

$$v(G) = e - n + (2 \cdot p) \quad (5.1)$$

A Equação 5.1 foi desenvolvida por McCabe (1976) e ela foi aplicada nessa tese para atribuir pesos às conexões entre os componentes dos modelos computacionais. Essa equação tem o nome de Complexidade Ciclomática (CC) e ela é uma das métricas mais utilizada na mensuração da complexidade de acordo com Polančič e Cegnar (2017). A CC é uma ferramenta que auxilia na medição da complexidade da estrutura de fluxo de controle de um sistema. Em termos práticos, conforme McCabe (1976), a CC mede o poder de tomada de decisão lógica de um sistema. Na Equação 5.1, o termo $v(G)$ é o número ciclomático e significa quantas decisões lógicas são tomadas no sistema; o termo n representa o número de vértices do sistema; o termo e significa a quantidade de arestas do sistema; o termo p representa a quantidade de blocos de códigos conectados do sistema. Por exemplo, considere a seguinte conjunto de linhas de código:

```

182 | Value_T_Op01 = value;
183 |
184 | if (Value_T_Op01 == 0){
185 |
186 |     Estados_Op01 = "Descansando";
187 |     Horario_Op01 = Op01.getHourOfDay();
188 |
189 | }else if (Value_T_Op01 == 2){
190 |
191 |     Estados_Op01 = "Almocando";
192 |     Horario_Op01 = Op01.getHourOfDay();
193 |
194 | }else{
195 |
196 |     Estados_Op01 = "Trabalhando";
197 |     Horario_Op01 = Op01.getHourOfDay();
198 |
199 | }
```

Esse conjunto de linhas de código representa a conexão do componente **Transicao_1** com os componentes **Value_T_Op01**, **Estados_Op01**, **Horario_Op01** e **Op01**. A Figura 5.1 mostra o grafo desse conjunto de linhas de código.

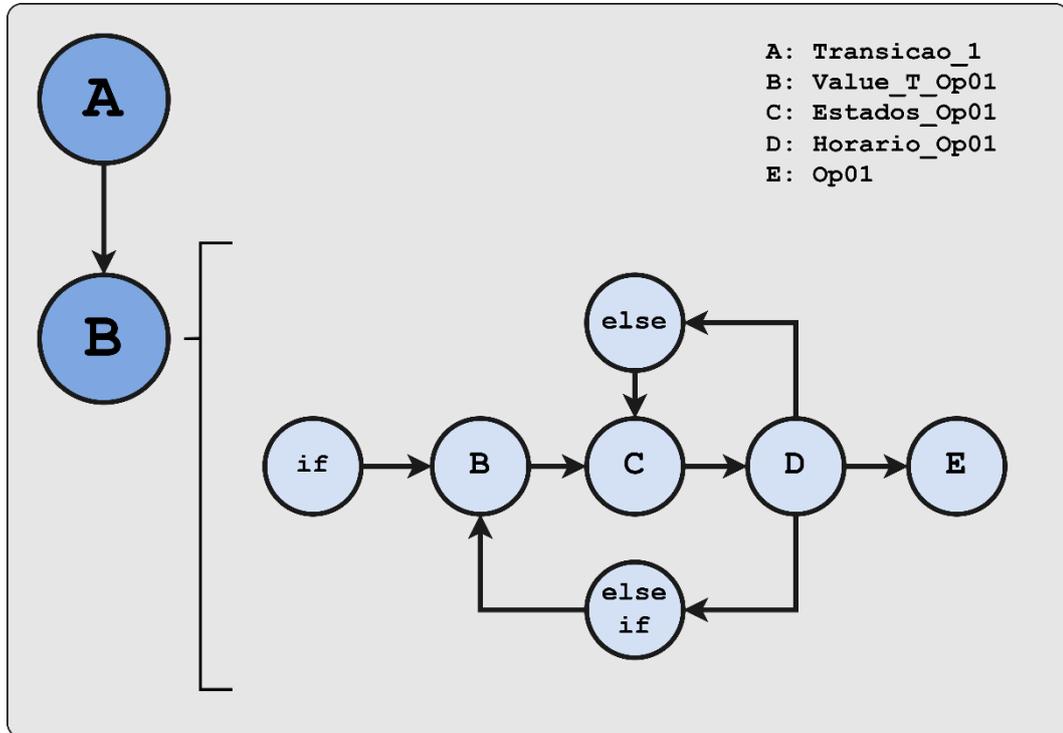


Figura 5.1 - Grafo das linhas de código 182 a 199
 Fonte: Elaborado pelo próprio autor

Com base na Figura 5.1, pode-se aplicar a Equação 5.1. O resultado é:

$$v(G) = e - n + (2 \cdot p) = 9 - 9 + (2 \cdot 2) = 4 \quad (5.2)$$

Com a Figura 5.1 e a Equação 5.2 fica claro a existência de dois blocos de código. Sendo a interseção, a variável **Value_T_Op01** coleta o valor do calendário **Transicao_1**. Esse é o primeiro bloco. O valor desse bloco é testado pela lógica *if – else if – else* que verifica qual o valor da variável **Value_T_Op01** e determina as ações para as variáveis **Estados_Op01** e **Horario_Op01**. Esse é o segundo bloco. Portanto, de modo simplificado, a função p da Equação 5.1 representa o número de sub-rotinas existentes em um conjunto de códigos.

A aplicação da CC ocorreu para todas as linhas de código existentes nos modelos DON3, H0N3, D1N3 e H1N3. Nesses modelos computacionais, as conexões do tipo *drag-and-drop* receberam peso igual a 1, seguindo a lógica da ferramenta CC. A Tabela 5.2 resume os resultados da CC encontrados para as conexões existentes dentro de cada modelo computacional.

Tabela 5.2 - Detalhamento dos pesos atribuídos às conexões de cada modelo

$v(G)$	D0N3	H0N3	D1N3	H1N3
1	24	78	108	378
4	12	0	72	0
Total	36	78	180	378

Fonte: Elaborado pelo próprio autor

Após as etapas de **determinação do critério de pesos** e **atribuição de pesos**, iniciou-se a etapa de **determinação do método de layout do grafo**. Com base nos trabalhos de Godehardt (1988), Di Battista *et alii* (1994), Herman, Melançon e Marshall (2000), Kaufmann e Wagner (2001), Koren (2005), Wallis (2007), Jacomy *et alii* (2014) e Tamassia (2014), os principais métodos de layout de grafos são: *layout em árvore*, *layout em camadas*, *layout circular*, *layout baseado em arco*, *layout de dominância*, *layout espectral*, *layout baseado em força* e *layout ortogonal*. Nessa tese, foram escolhidos dois layouts: o (i) layout baseado em força e o (ii) layout circular. A escolha teve como base a ênfase gráfica dada por esses layouts. No layout baseado em força, a ênfase gráfica está sobre o grau de importância de cada componente dentro do sistema estudado. No layout circular, a ênfase está no grau de relação entre os componentes dentro do sistema estudado. Assim, esses layouts vão auxiliar no processo de análise estrutural dos modelos computacionais, pois os grafos geram resultados visuais que facilitam o entendimento sobre os modelos computacionais e suas particularidades. Na próxima seção, serão descritas as etapas da fase **Análise**.

5.3. Análise

A fase **Análise** possui duas etapas: (i) **criação do grafo e parâmetros estéticos**, e (ii) **inferência sobre os resultados**. Assim, iniciou-se o processo de criação dos grafos. Representando os modelos computacionais, os grafos foram construídos a partir de três dados: componentes, conexões e natureza das conexões. O número de componentes e o número de conexões de cada modelo computacional estão expressos na Tabela 5.1. A natureza das conexões existentes em cada modelo computacional está explícito na Tabela 5.2.

Para construir os grafos, três procedimentos foram necessários. O primeiro foi a criação de uma matriz representando o número de componentes, as conexões entre esses componentes e a natureza dessas conexões. O segundo procedimento foi a transferência da matriz de cada modelo computacional para o software Gephi® – software de código aberto utilizado para

análise e visualização de redes. O terceiro foi o procedimento de configuração das cores e dos tipos de grafos. As Figuras 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8 e 5.9 mostram os resultados alcançados com base nos três procedimentos.

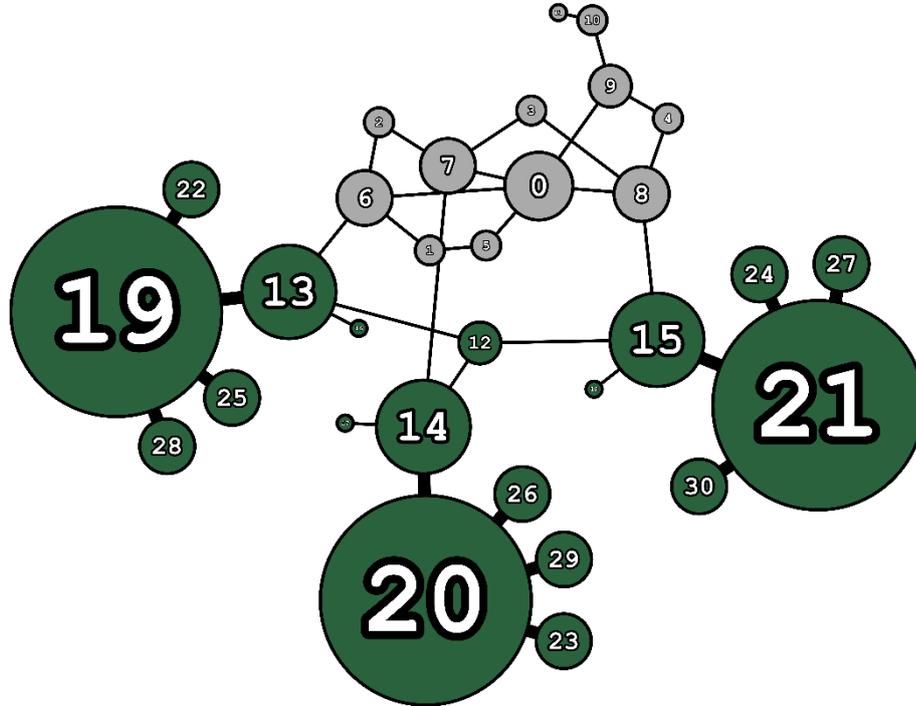


Figura 5.2 - Grafo de estrutura do modelo D0N3
Fonte: Elaborado pelo próprio autor

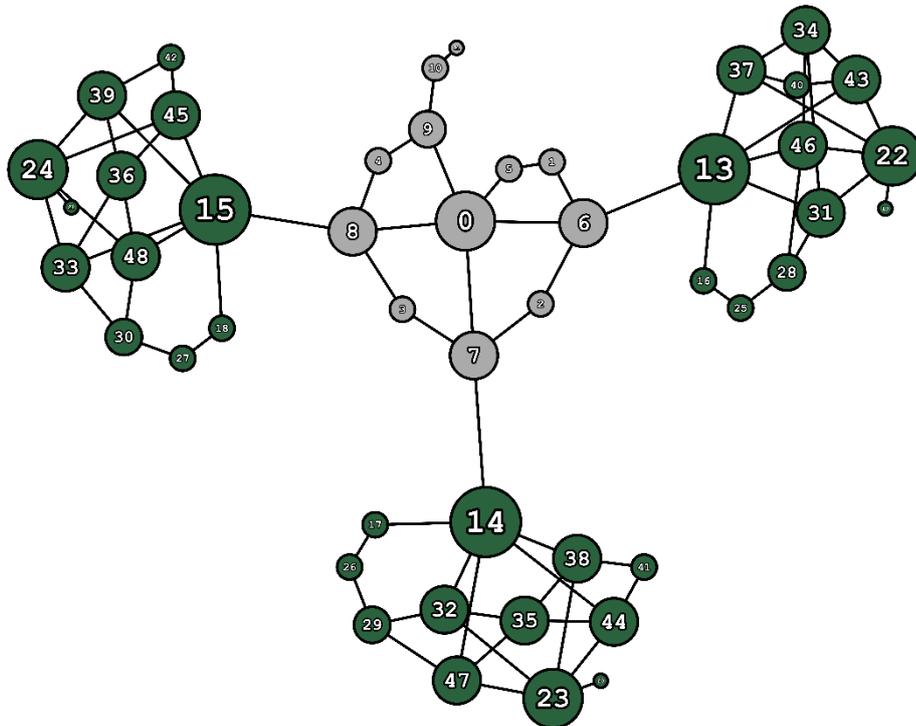


Figura 5.3 - Grafo de estrutura do modelo H0N3
Fonte: Elaborado pelo próprio autor

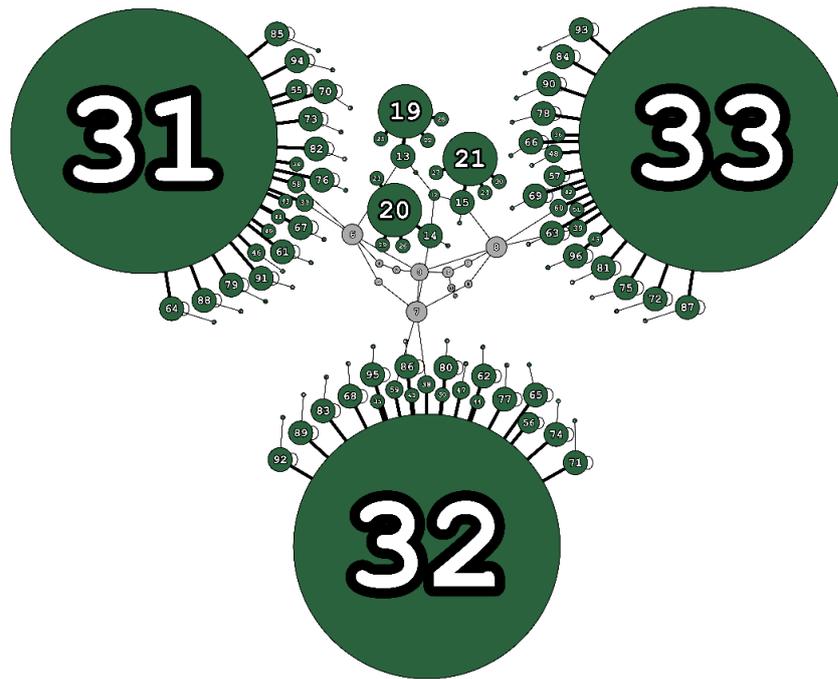


Figura 5.4 - Grafo de estrutura do modelo D1N3
 Fonte: Elaborado pelo próprio autor

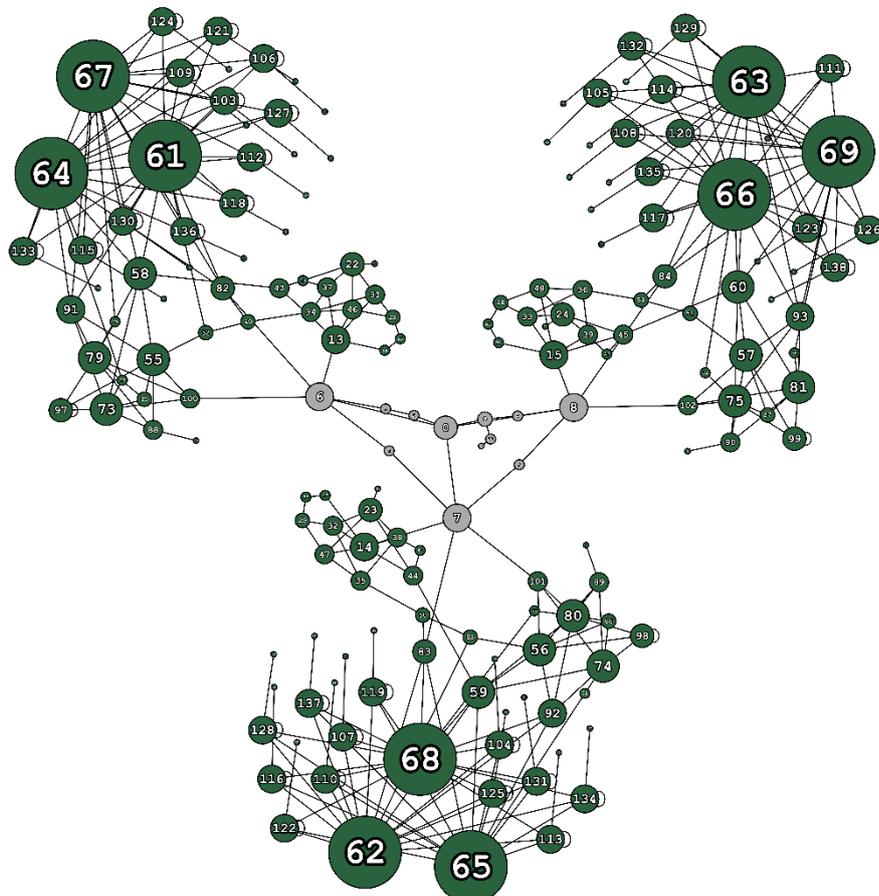


Figura 5.5 - Grafo de estrutura do modelo H1N3
 Fonte: Elaborado pelo próprio autor

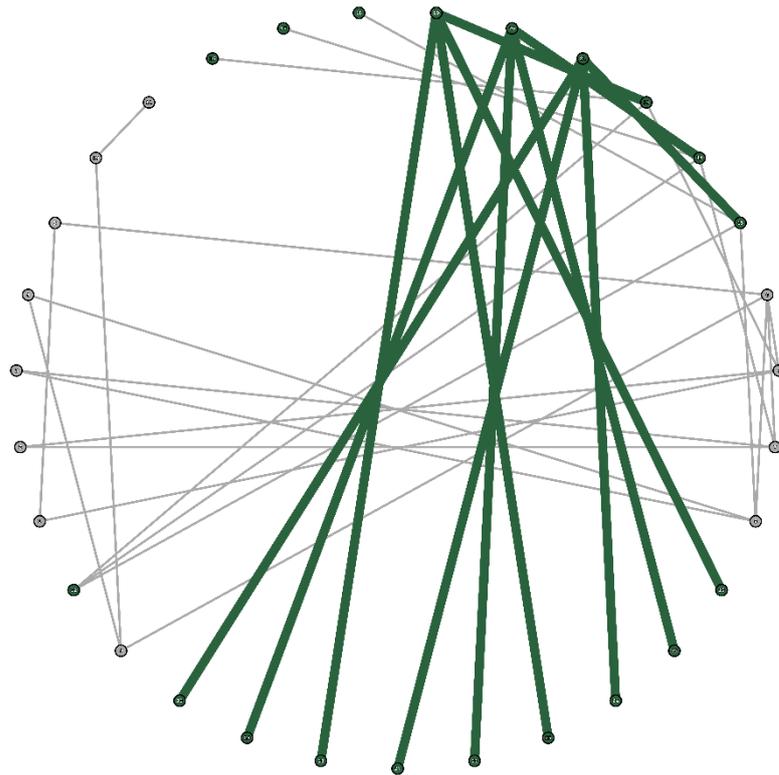


Figura 5.6 - Grafo de essência do modelo D0N3

Fonte: Elaborado pelo próprio autor

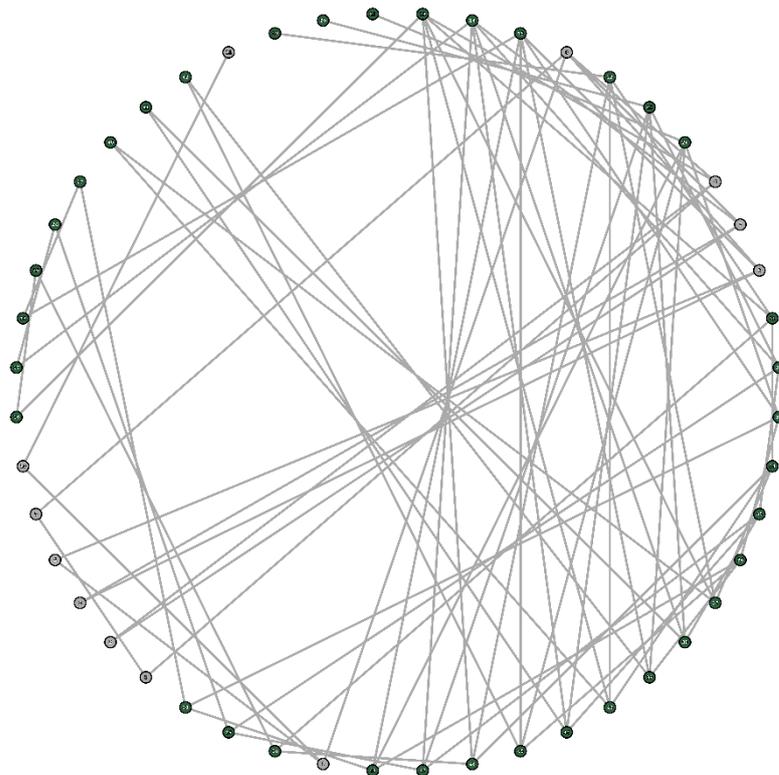


Figura 5.7 - Grafo de essência do modelo D0N3

Fonte: Elaborado pelo próprio autor

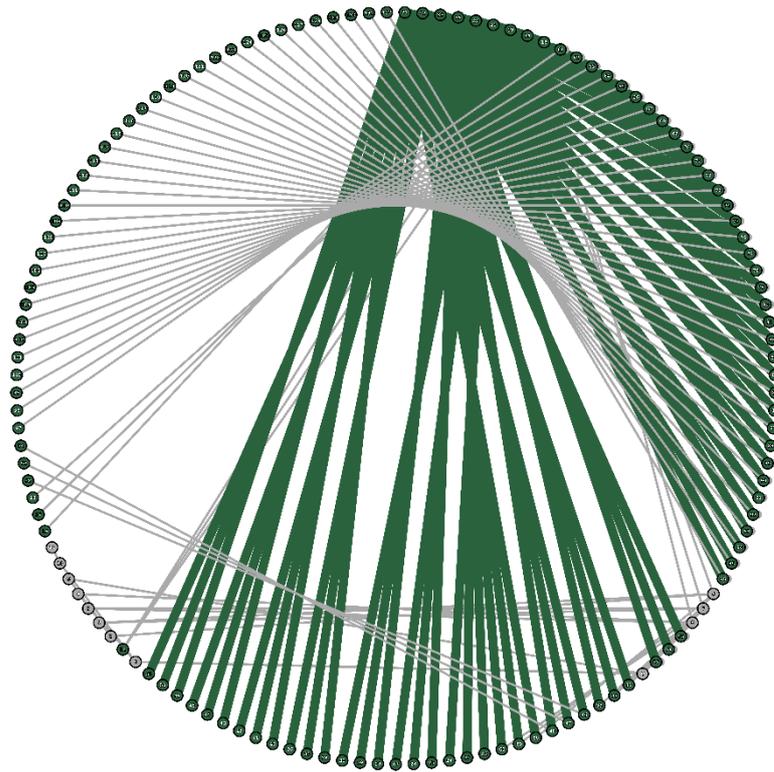


Figura 5.8 - Grafo de essência do modelo D1N3

Fonte: Elaborado pelo próprio autor

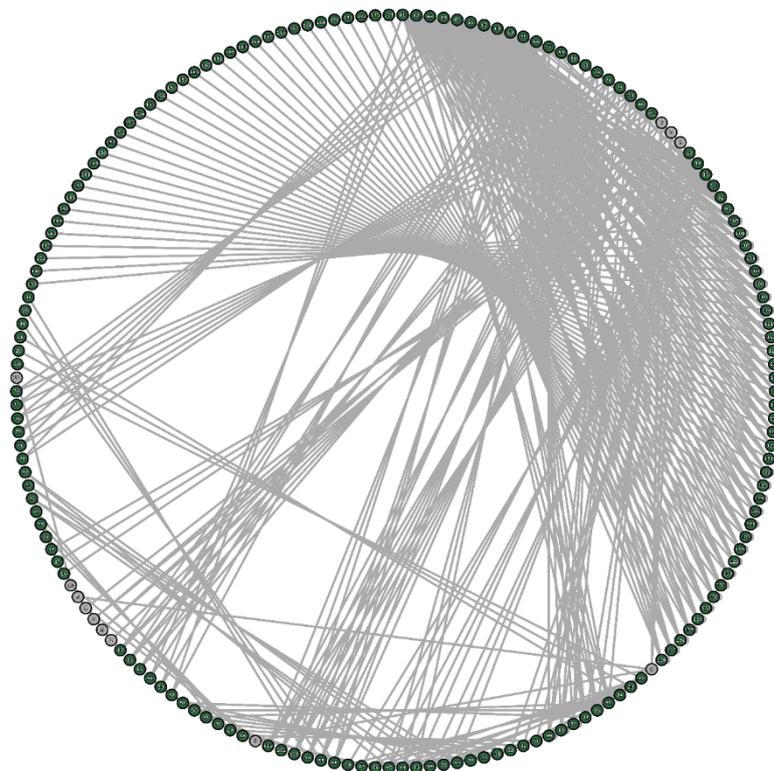


Figura 5.9 - Grafo de essência do modelo H1N3

Fonte: Elaborado pelo próprio autor

As Figuras 5.2, 5.3, 5.4 e 5.5 são grafos de estrutura que representam, respectivamente, os modelos D0N3, H0N3, D1N3 e H1N3. A ênfase dos grafos de estrutura está na estrutura dos modelos computacionais na qual é possível separá-la em dois setores: Fluxo de Processo (FP) e Comportamento da Entidade (CE). Nos grafos estruturais dos modelos, o setor FP e CE estão destacados, por essa ordem, em cinza e verde. As Figuras 5.6, 5.7, 5.8 e 5.9 são denominados grafos de essência e eles retratam, respectivamente, os modelos D0N3, H0N3, D1N3 e H1N3. Os grafos de essência enfatizam a natureza das conexões existentes entre os componentes de um modelo. Por exemplo, uma determinada conexão entre dois componentes é enfatizada por meio da cor e da espessura da linha. A cor e a espessura da linha refletem o peso que a conexão possui. Ademais, cada componente possui um número de identificação e esse número está explícito nos grafos.

Com os grafos criados, o cálculo do Grau Médio Ponderado (GMP) foi efetuado. Cada componente possui um número que determina o seu grau. Por exemplo, o componente **Fim** – componente conectado aos componentes **connector4**, **Peca** e **TotalProduzido** –, possui grau igual a 3. Isso significa que, no total, 3 conexões de peso 1 estão vinculadas ao componente **Fim**. O GMP representa a soma dos graus dos componentes do modelo dividido pela quantidade total de componentes do modelo. O resumo dos resultados se encontra na Tabela 5.3.

Tabela 5.3 - Grau médio ponderado dos modelos

Modelos computacionais	Σ Graus	Componentes	GMP
D0N3	144	31	4,65
H0N3	156	48	3,25
D1N3	792	133	5,96
H1N3	756	174	4,35

Fonte: Elaborado pelo próprio autor

Os resultados expressos nas Tabelas 5.1, 5.2 e 5.3, e das Figuras 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8 e 5.9 serão discutidas nessa etapa de inferência sobre os resultados. Com base nos dados da Tabela 5.1, quando se compara o número de componentes entre os modelos D0N3 e D1N3, nota-se que houve um aumento na ordem de 329%. Observa-se também que, quando se compara os modelos H0N3 e H1N3, o aumento foi de 263%. Esse aumento ocorre em função dos novos componentes inseridos nos modelos do segundo nível de detalhamento, em particular, os componentes referentes à Tabela 4.1.

Seguindo a mesma circunstância, o número de conexões do modelo D1N3 é 400% maior do que o número de conexões do modelo D0N3. No caso dos modelos H0N3 e H1N3, o número de conexões do H1N3 é 385% maior do que o número de conexões do modelo H0N3. O valor de 385% é consequência do aumento de 86% do número de conexões sem linha de código. Esse aumento não ocorre quando se compara o número de conexões sem linha de código dos modelos D0N3 e D1N3. No caso do número de conexões com linha de código, houve aumento de 1029%, quando se compara os modelos D0N3 e D1N3, e de 890%, quando se compara os modelos H0N3 e H1N3. Esses dois últimos aumentos ocorrem, principalmente, devido ao acréscimo de maior autonomia das entidades/agentes dentro dos modelos. A Tabela 5.4 mostra o resumo dos aumentos.

Tabela 5.4 - Comparação entre modelos que utilizam a mesma abordagem de simulação

Modelos computacionais	Nº de componentes	Nº de conexões		
		<i>Sem linha de código</i>	<i>Com linha de código</i>	<i>Total</i>
D0N3 vs. D1N3	329%	0%	1029%	400%
H0N3 vs. H1N3	263%	86%	890%	385%

Fonte: Elaborado pelo próprio autor

Tabela 5.5 - Comparação entre modelos que utilizam abordagens de simulação diferentes

Modelos computacionais	Nº de componentes	Nº de conexões		
		<i>Sem linha de código</i>	<i>Com linha de código</i>	<i>Total</i>
D0N3 vs. H0N3	55%	123%	107%	117%
D1N3 vs. H1N3	31%	314%	82%	110%

Fonte: Elaborado pelo próprio autor

Ainda com base na Tabela 5.1, observa-se, na Tabela 5.5, que o modelo H0N3 possui 55% de componentes a mais do que o modelo D0N3. No caso da comparação entre os modelos H1N3 e D1N3, o aumento é de 31%. Esses aumentos são naturais, pois, devido ao uso de diagramas de estados, o número de componentes tende a aumentar na abordagem híbrida. Conseqüentemente, o número de conexões também aumenta. Isso é constatado quando se observa que o número de conexões do modelo H0N3 é 117% maior do que o número de conexões do modelo D0N3. Por outro lado, o número de conexões do modelo H1N3 é 110% maior do que o número de conexões do modelo D1N3. O valor 110% é explicado pelo valor 314% que representa o aumento de números de conexões sem linha de código entre os

modelos H1N3 e D1N3. No caso das conexões com linha de código, o modelo H0N3 possui 107% de linhas a mais do que o modelo D0N3. Já a comparação entre os modelos H1N3 e D1N3 mostra o aumento de 82%. Todos esses resultados estão resumidos na Tabela 5.5.

Outro resultado que pode ser retirado da Tabela 5.1 é a proporção de conexões sem linhas de código e com linhas de código de cada modelo computacional. Nota-se que 61% das conexões existentes no modelo D0N3 são conexões sem linhas de código. No modelo H0N3, o valor é de 63%. Esses percentuais são menores nos modelos D1N3 e H1N3, respectivamente, na ordem de 12% e de 24%. A Tabela 5.6 mostra os valores percentuais para cada modelo.

Tabela 5.6 - Proporção dos tipos de conexões de cada modelo computacional

Modelos computacionais	Nº de conexões		
	Sem linha de código	Com linha de código	Total
D0N3	61%	39%	100%
H0N3	63%	37%	100%
D1N3	12%	88%	100%
H1N3	24%	76%	100%

Fonte: Elaborado pelo próprio autor

Com base nos dados da Tabela 5.2, pode-se detalhar os valores percentuais da Tabela 5.6 conforme os pesos atribuídos às conexões dos modelos D0N3, H0N3, D1N3 e H1N3. A Tabela 5.7 mostra esse detalhamento.

Tabela 5.7 - Detalhamento percentual dos pesos atribuídos às conexões de cada modelo

$v(G)$	D0N3	H0N3	D1N3	H1N3
1	67%	100%	60%	100%
4	33%	0%	40%	0%
Total	100%	100%	100%	100%

Fonte: Elaborado pelo próprio autor

Até o momento, com os resultados coletados, pode-se fazer as seguintes afirmações:

- I. O número de componentes e conexões aumentam quando se aumenta o grau de autonomia das entidades/agentes nos modelos computacionais;
- II. O aumento do grau de autonomia das entidades/agentes gera o crescimento de conexões com linhas de código;

- III.** Há o aumento no número de componentes e de conexões quando se utiliza a abordagem SH;
- IV.** Os modelos H0N3 e H1N3 possuem um nível de peso. Por outro lado, os modelos D0N3 e D1N3 possuem dois níveis de pesos.

Sobre a afirmação **I**, a construção de modelos com entidades/agentes mais autônomos pressupõe a utilização de mais parâmetros e variáveis. Portanto, independentemente do tipo de abordagem de simulação utilizada, a quantidade de componentes e conexões irão aumentar. Esse fato é evidenciado na Tabela 5.4.

Sobre a afirmação **II**, as conexões com linhas de código são inferiores às conexões sem linhas de código nos modelos em que as entidades/agentes têm menos autonomia. Esse cenário se inverte para os modelos em que as entidades/agentes têm mais autonomia. Essa inversão se dá por dois fatores: (i) alteração no grau de autonomia e (ii) alteração na abordagem de simulação. Isso se torna evidente quando se observa as Tabelas 5.5 e 5.6.

Sobre a afirmação **III**, a principal razão do aumento de componentes e conexões está na utilização de diagramas de estados – ferramenta existente na abordagem SBA. Os diagramas de estados substituem os blocos de programação existentes nos modelos que utilizam a abordagem SED e, do ponto de vista da programação, simplificam a construção dos modelos computacionais. Entretanto, o efeito colateral é o aumento no número de componentes e conexões.

Sobre a afirmação **IV**, percebe-se que o uso de diagramas de estados impede a programação de blocos que utilizam a estrutura lógica *if – else if – else*. Isso faz com que os modelos de SH possuam menos níveis de pesos do que os modelos de SED.

Considerando os pesos atribuídos em cada conexão, a Tabela 5.3 expõe a soma dos graus dos componentes de cada modelo. Observa-se que o valor do GMP do modelo D0N3 é maior do que o valor do GMP do modelo H0N3. O mesmo ocorre entre os modelos D1N3 e H1N3. O GMP é um parâmetro que pode servir para compreender, em termos numéricos, a complexidade dos modelos computacionais. Por exemplo, comparando os modelos D0N3 e D1N3, houve aumento no valor do GMP em razão da atribuição de maior autonomia às entidades/agentes. A mesma constatação pode ser aferida quando se compara os modelos H0N3 e H1N3.

Sobre as duas abordagens de simulação, SED e SH, nota-se também a ausência de similaridade nos valores do GMP. Esses valores são efeitos da forma de modelagem. *Verbi*

gratia, para produzir as mesmas resposta e informações, os modelos D0N3 e D1N3 possuem códigos baseados na lógica *if – else if – else*. Nos modelos H0N3 e H1N3, essa lógica ela também se encontra presente, porém na forma de diagrama de estado.

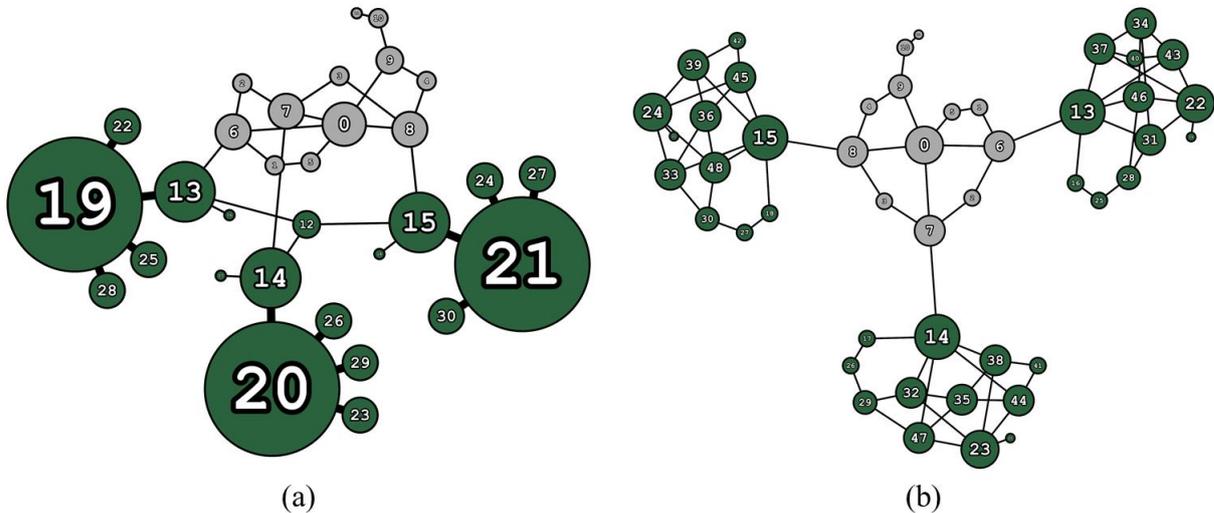


Figura 5.10 - Grafos de estrutura dos modelos D0N3 (a) e H0N3 (b)
Fonte: Elaborado pelo próprio autor

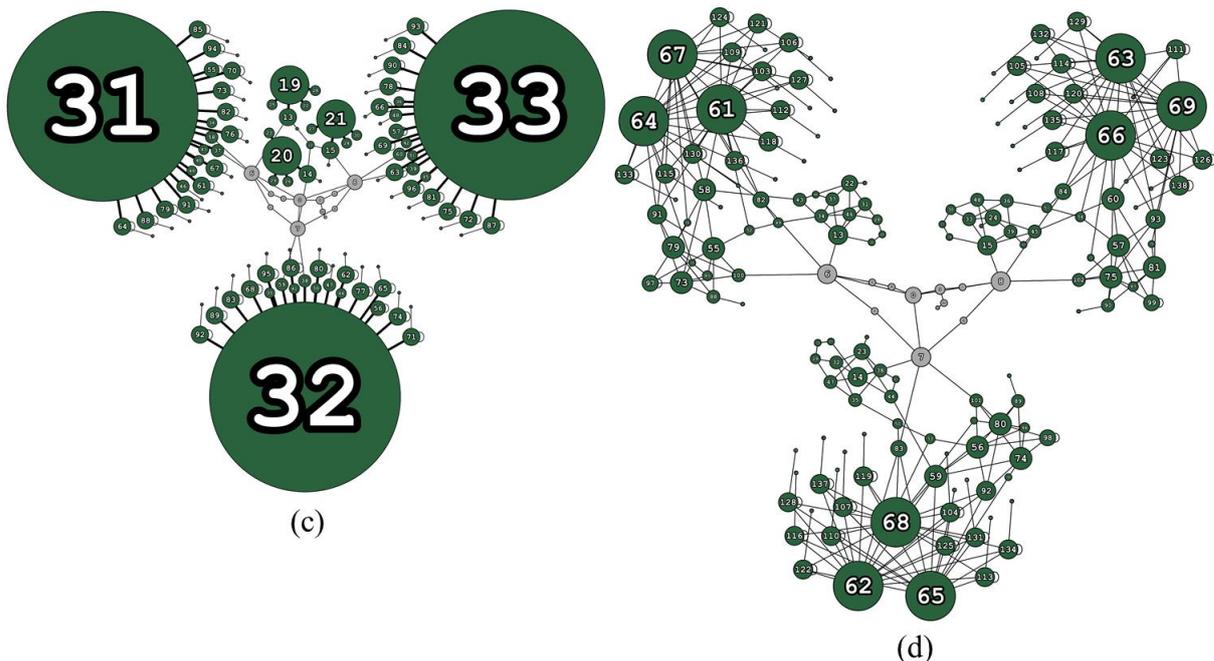


Figura 5.11 - Grafos de estrutura dos modelos D1N3 (c) e H1N3 (d)
Fonte: Elaborado pelo próprio autor

As Figuras 5.10 e 5.11 retratam a comparação entre os grafos de estrutura dos modelos computacionais. Cabe salientar que os grafos (a), (b), (c) e (d) representam, respectivamente, os modelos D0N3, H0N3, D1N3 e H1N3. Como afirmado anteriormente, o setor FP e CE estão destacados, por essa ordem, em cinza e verde. O setor FP possui 12 componentes, sendo esse número o mesmo em todos os modelos computacionais. No setor CE, no qual encontra-

se as idiossincrasias das entidades/agentes, há mudanças significativas quanto ao número de componentes. Considerando o setor CE, os modelos D0N3, H0N3, D1N3 e H1N3 possuem, respectivamente, 19, 36, 121 e 162 componentes.

Visualmente, existem componentes cujo diâmetro é maior. O diâmetro de um dado componente é alterado conforme a quantidade de conexões nele presente e os pesos atribuídos a essas conexões. Neste ponto, ao representar um modelo computacional através de grafos, é possível identificar quais são os componentes essenciais dentro do modelo. Por exemplo, na Figura 5.10, no grafo (a) destacam-se os componentes 19, 20 e 21. Esses componentes estão vinculados ao comportamento roteirizado das entidades/agentes no modelo D0N3. Na mesma figura, grafo (b), percebe-se uma quantidade maior de componentes e conexões que também estão vinculados ao comportamento roteirizado das entidades/agentes no modelo H0N3.

O uso da abordagem SH produz o efeito de pulverização. Na realidade, o diagrama de estado, intrínseco à abordagem SBA, faz com que exista um aumento na quantidade de componentes em um modelo computacional e uma diminuição na complexidade das conexões entre os componentes. Tal diminuição se dá em razão da simplificação da escrita das linhas de código. Os códigos baseados na lógica *if – else if – else* encontra-se na estrutura do diagrama de estado e os comandos mais simples estão nas transições de um estado para o outro. Tudo isso fica evidente quando na Figura 5.11.

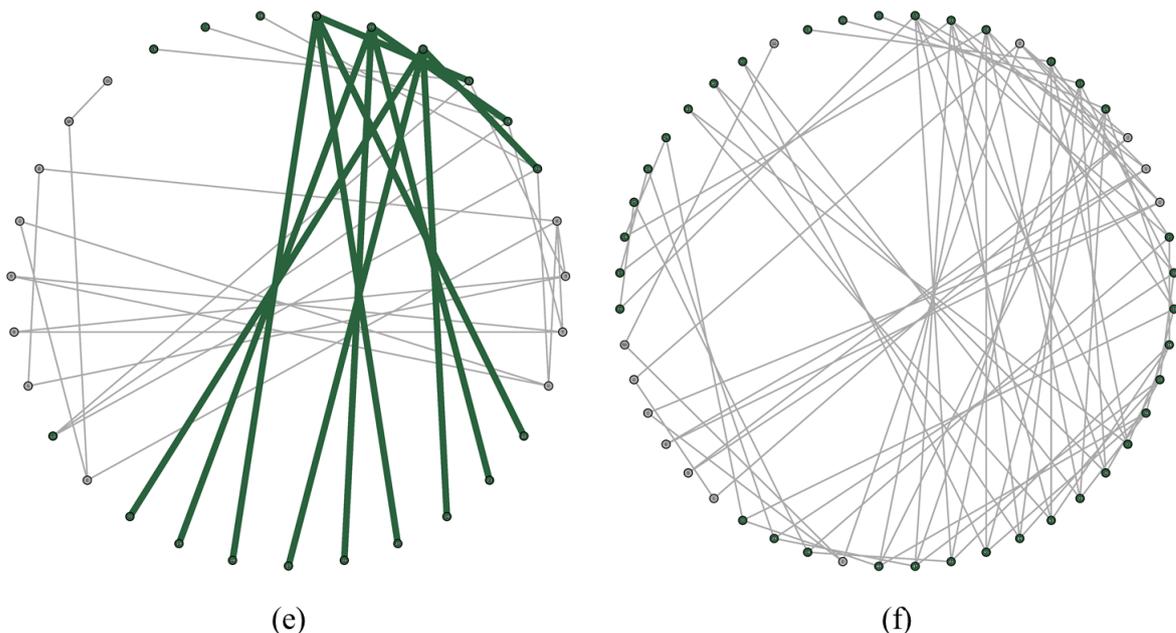


Figura 5.12 - Grafos de essência dos modelos D0N3 (e) e H0N3 (f)

Fonte: Elaborado pelo próprio autor

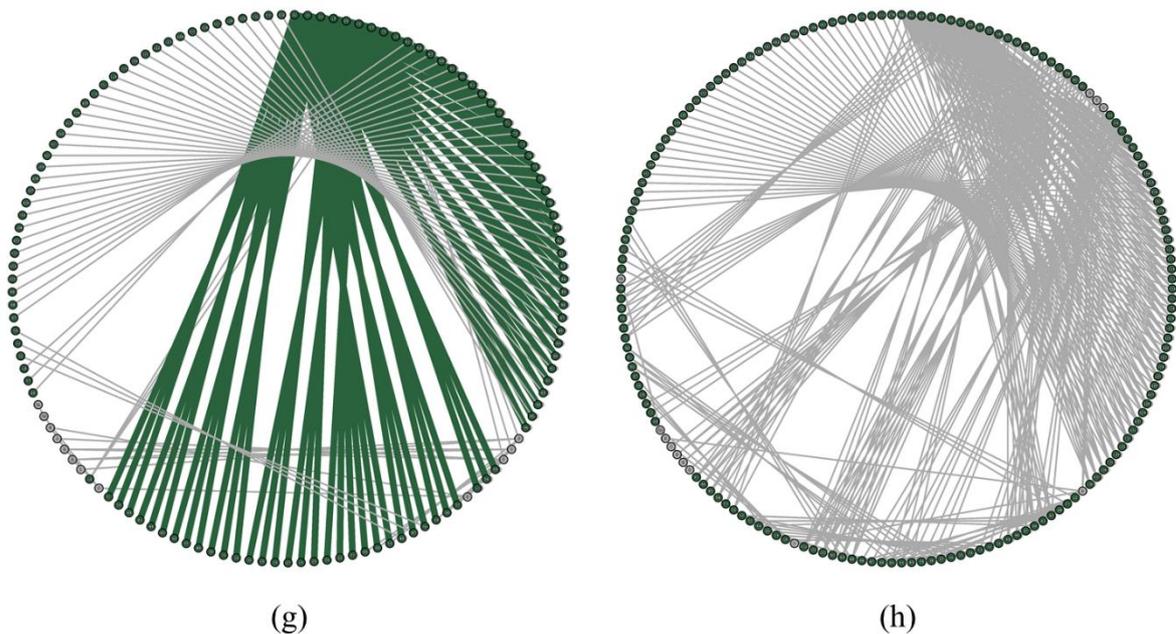


Figura 5.13 - Grafos de essência dos modelos D1N3 (g) e H1N3 (h)
 Fonte: Elaborado pelo próprio autor

Além dos grafos de estrutura, essa tese faz uso de grafos de essência. Nesses grafos, a ênfase se dá sobre as conexões entre os componentes e a natureza de cada uma delas. É por essa razão que os componentes possuem o mesmo diâmetro. Os grafos de essência possuem a característica de serem circulares. Os componentes estão ordenados do maior para o menor, conforme o seu grau ponderado. Esse é o motivo que faz com que as conexões em destaque se concentrem no lado direito dos grafos; isso fica mais evidente nos grafos de essência da Figura 5.13. Ademais, as espessuras e as cores das linhas dos grafos de essência retratam os valores de $\nu(G)$ das conexões, ou seja, os pesos atribuídos às linhas. Assim, as linhas cujos pesos são maiores possuem espessura maior e coloração em verde.

Os grafos (e) e (f) da Figura 5.12 representam, respectivamente, os modelos D0N3 e H0N3. Ao comparar esses grafos, nota-se que o grafo (e) possui linhas mais espessas e escuras do que o grafo (f). Essas linhas possuem valores de $\nu(G)$ maiores em razão dos códigos baseados na lógica *if – else if – else*. O mesmo efeito pode ser constatado por meio da comparação entre os grafos (g) e (h) da Figura 5.13.

Os grafos (f) e (h), que retratam os modelos criados pela abordagem SH, possuem mais linhas e componentes. No caso desses grafos, não há linhas espessas e com coloração verde. Como afirmado anteriormente, o diagrama de estado inibe a construção de blocos de códigos baseados na lógica *if – else if – else*, portanto, na abordagem SH, existe a tendência de se

programar com comandos mais simples. Contudo, na abordagem SH, a quantidade de componentes é maior, bem como a quantidade de conexões.

Nessa seção o segundo objetivo específico foi atingido. Os grafos apresentados nessa tese auxiliam na compreensão e na análise da estrutura/essência dos modelos computacionais. Destaca-se o interessante efeito visual dos grafos que acaba por explicitar as diferenças intrínsecas quanto ao uso das abordagens SED e SH na criação dos modelos. Também pode ser salientado o efeito visual que entidades/agentes, e seus graus de autonomia, geram nos grafos de estrutura ou essência.

5.4. Verificação

A fase de verificação contém três etapas: (i) **verificação de erros na fase identificação**, (ii) **verificação de erros na fase definição** e (iii) **verificação de erros na fase análise**. O método iDAV funciona de maneira cíclica, visando o aprimoramento contínuo do processo de mensuração da complexidade dos modelos computacionais. Dessa forma, houve cinco ciclos de verificação. Esses ciclos, além de eliminar erros encontrados nas fases anteriores, ajudaram a aprimorar os modelos computacionais de modo a torná-los mais enxutos em termos de linhas de códigos e de componentes.

Antes do processo de mensuração de complexidade houve a aplicação do IDEF-SIM e dos diagramas de estados no desenvolvimento dos modelos conceituais. O desafio desse processo foi a exposição de maneira clara dos elementos importantes para a construção dos modelos computacionais. O ponto crítico dos modelos conceituais está na ausência de recursos capazes de evidenciar todas as variáveis e os parâmetros necessários para o desenvolvimento dos modelos computacionais.

Com os modelos conceituais concluídos, iniciou-se o processo de construção dos modelos computacionais com as abordagens SED e SH. Esses modelos passaram pelo processo de validação composto por cinco etapas, como mostra a Figura 4.6. Foi nesse ponto em que os modelos computacionais de SED e de SH foram comparados estatisticamente. Os primeiros resultados do processo de validação foram negativos, *id est*, os modelos computacionais de abordagens diferentes não eram estatisticamente iguais. Então, os modelos computacionais passaram por um processo de aprimoramento onde foram identificados erros quanto ao número de componentes e conexões. No caso dos modelos de SED, reduziu-se ao máximo o número de linhas de código, houve inserção de calendários e de variáveis que descrevem os estados dos operadores e ocorreu uma revisão geral da lógica que rege os modelos. No caso

dos modelos de SH, houve uma análise crítica sobre os diagramas de estados, houve também a inserção de variáveis e, por fim, ocorreu uma análise quanto à lógica geral dos modelos. Após o processo de avaliação crítica dos modelos, os mesmos foram validados estatisticamente. Vale ressaltar que esse processo de avaliação crítica dos modelos ocorreu em diversos momentos no decorrer de todas as fases do método iDAV.

Com os modelos computacionais validados estatisticamente, iniciou-se a fase *identificação*. Nessa etapa, aconteceu a contagem dos componentes e das conexões existentes entre esses componentes. Devido à ausência de ferramentas mais apropriadas, esse processo de contagem ocorreu de maneira manual e, por isso, esse processo se repetiu algumas vezes para garantir a eliminação de erros. O resultado desse processo de contagem foi uma matriz para cada modelo computacional composta pelos componentes e pelas as conexões entre eles.

Na fase *Definição*, os pesos correspondentes de cada conexão foram inseridos dentro de cada matriz. Antes desse processo de inserção de pesos, foi necessário determinar o critério de pesos, e isso ocorreu a partir da Equação 5.1. Assim, cada conexão foi calculada e analisada de forma a verificar se aquele peso representava de fato a realidade. A escolha do método de layout do grafo ocorreu nessa fase. Essa escolha foi importante para determinar a ênfase de análise. Cabe ressaltar que outros métodos de layout foram testados, *verbi gratia*, layout baseado em arco e layout espectral. Porém, identificou-se que esses layouts não foram adequados o suficiente para exprimir, visualmente, as informações de estrutura e essência desejadas.

Na fase *Análise*, com base nas matrizes, os grafos foram criados por meio do software Gephi®. O processo de aprendizagem do pesquisador com relação a esse software foi relativamente baixo, pois o software é bem intuitivo. Entretanto, o processo de design dos grafos consome uma parte razoável de tempo. Com os grafos construídos, iniciou-se a etapa de inferência na qual comparações e análises foram desenvolvidas. Esse processo possibilitou a geração de conclusões interessantes tanto com relação aos resultados percentuais do número de componentes e de conexões de cada modelo, quanto com relação aos aspectos visuais dos grafos.

Na próxima seção, alguns testes de sensibilidade foram feitos para identificar os efeitos visuais, tanto nos grafos de estrutura quanto nos de essência, quando se acrescenta elementos gráficos (3D) e quando se acrescenta mais postos de trabalho na linha de produção.

6. AS ANÁLISES DE SENSIBILIDADE

Um dos objetivos dessa seção é analisar os efeitos gerados pela adição de elementos 3D. Em projetos de simulação, a adição de elementos 3D é interessante, pois melhora a comunicação com os *stakeholders*. Outro objetivo dessa seção é a análise dos efeitos gerados pelo aumento dos postos de trabalho. Até o momento, um sistema de manufatura com layout em linha foi utilizado. Esse sistema possui três postos de trabalho. O que ocorre se esse sistema triplicar?

6.1. Adição de elementos 3D

Um modelo de simulação deve transmitir, de maneira clara, informações assertivas de um sistema real. Em um projeto de simulação, a comunicação com os *stakeholders* é fundamental para que decisões não sejam tomadas de modo equivocado. É nesse ponto que a adição de elementos 3D se faz relevante.

Os elementos 3D estão intrinsecamente conectados com as estruturas lógicas do modelo computacional. Os elementos 3D complementam, visualmente, a comunicação e facilita o entendimento sobre o sistema real. Cabe salientar que a adição de elementos 3D gera um efeito negativo: o tempo de modelagem.

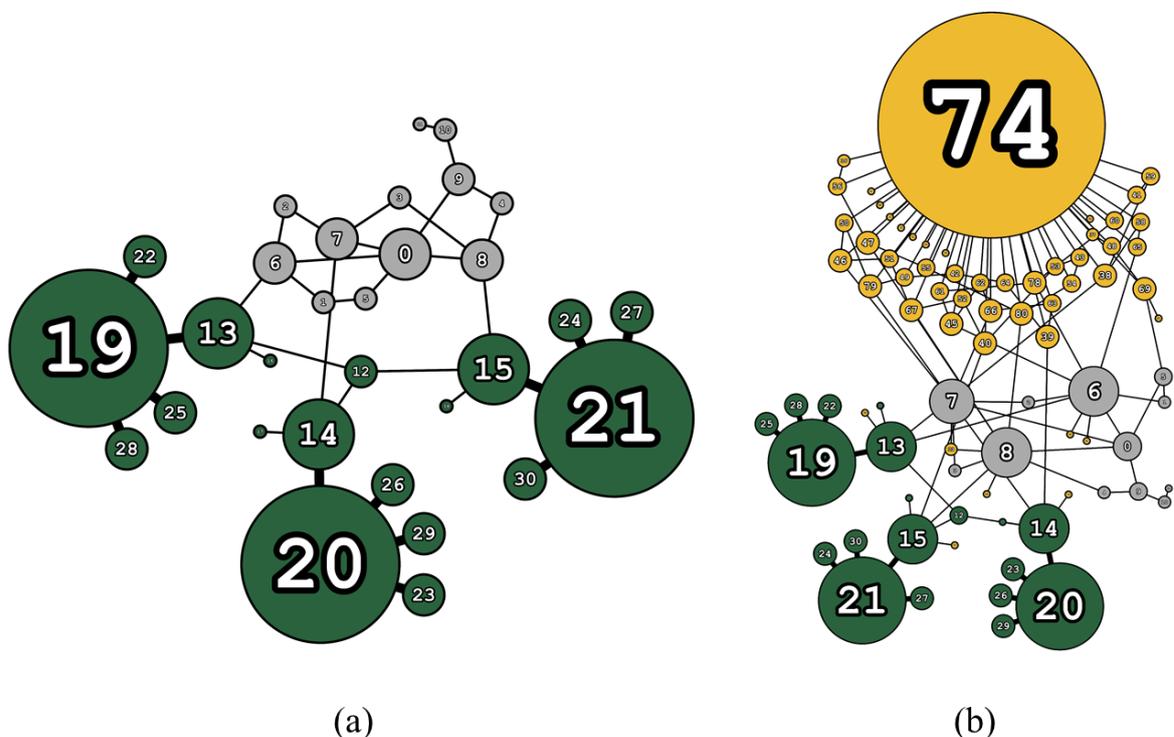


Figura 6.1 - Grafos de estrutura dos modelos D0N3 (a) e D0S3 (b)

Fonte: Elaborado pelo próprio autor

No total, foram adicionados 51 (cinquenta e um) elementos 3D. Essa adição ocorreu em todos os modelos de simulação. Assim, para diferenciá-los, os nomes dos modelos de simulação foram modificados, ficando: D0S3, H0S3, D1S3 e H1S3. Para melhor compreensão, *verbi gratia*, a letra **D** significa que o modelo computacional foi construído exclusivamente pela abordagem SED, o número **0** (**zero**) significa que os operadores não possuem características autônomas no modelo, a letra **S** significa que o modelo possui elementos relacionados à simulação 3D, e o número **3** (**três**) indica que o modelo possui três postos de trabalho.

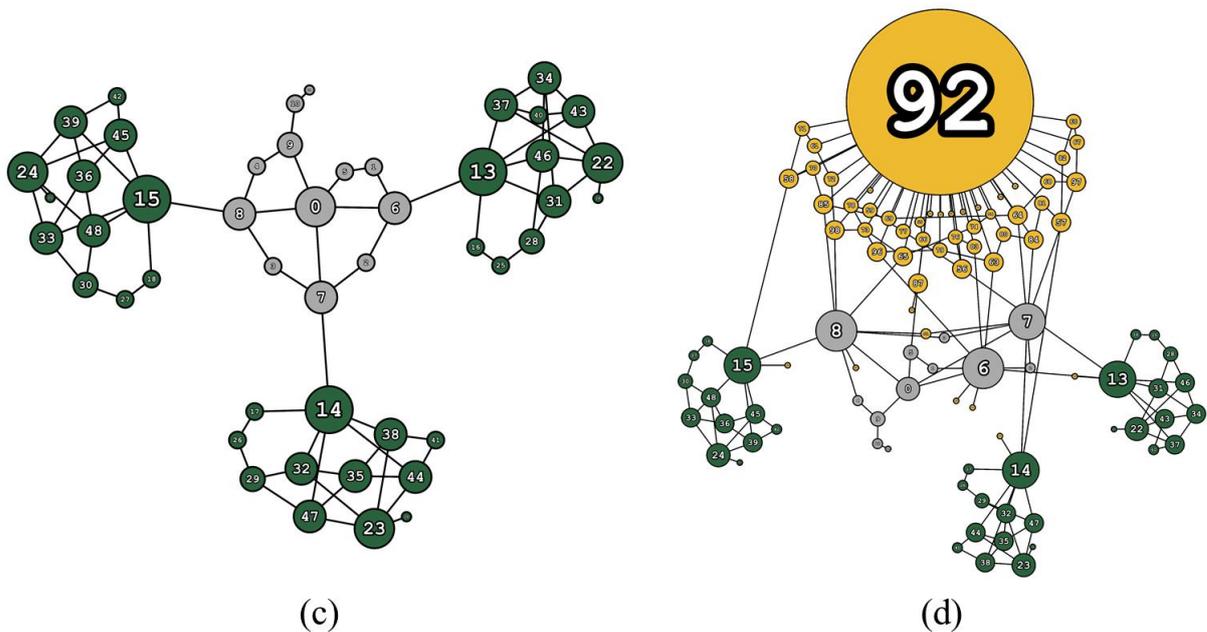


Figura 6.2 - Grafos de estrutura dos modelos H0N3 (c) e H0S3 (d)

Fonte: Elaborado pelo próprio autor

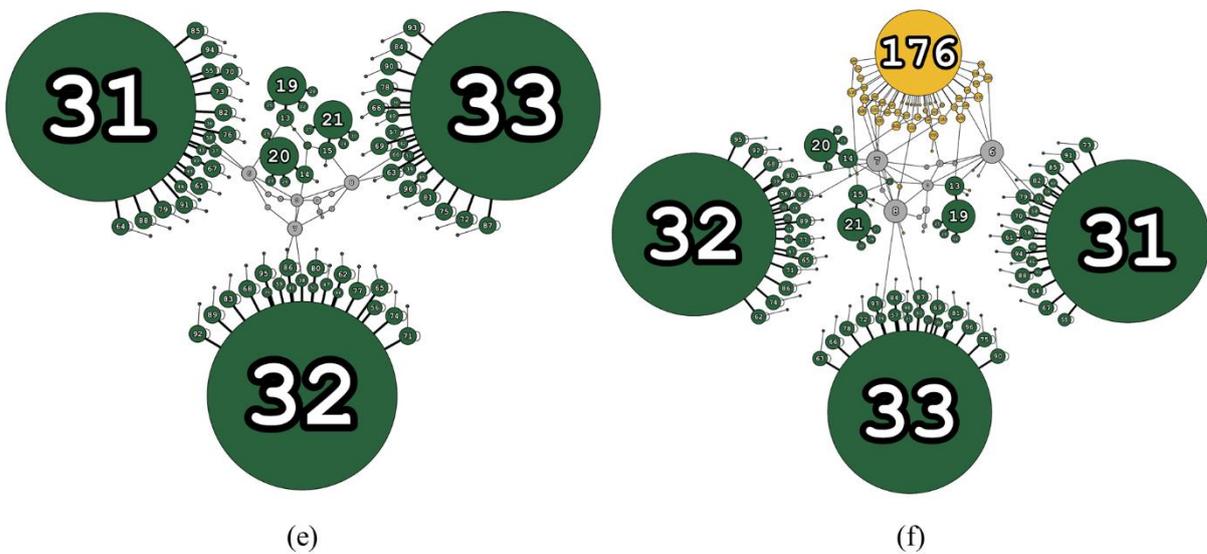


Figura 6.3 - Grafos de estrutura dos modelos D1N3 (e) e D1S3 (f)

Fonte: Elaborado pelo próprio autor

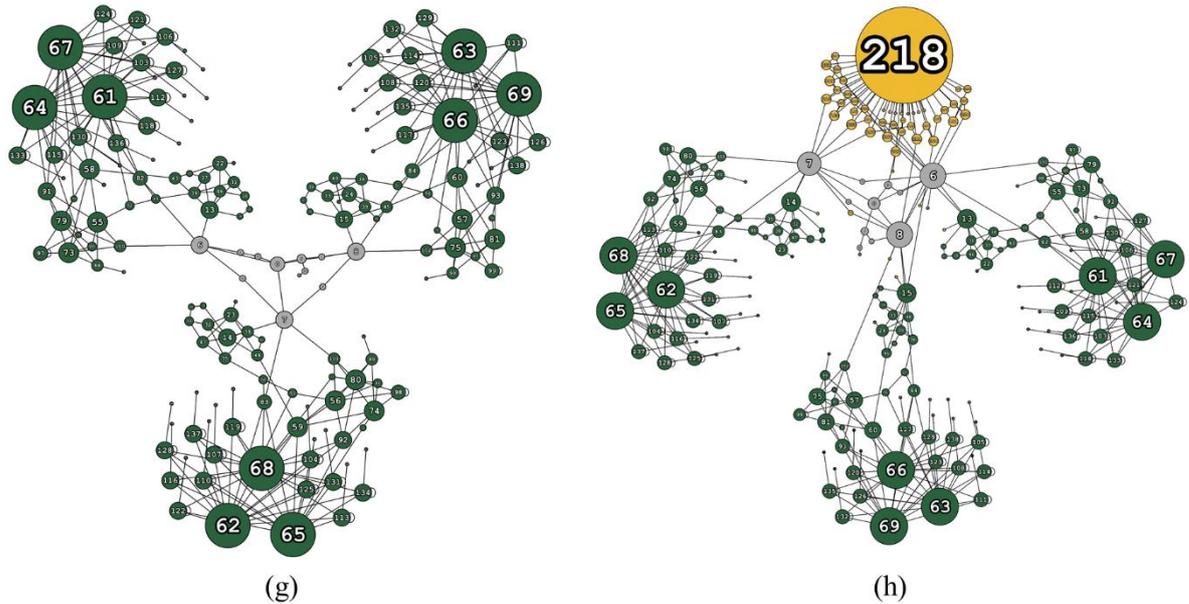


Figura 6.4 - Grafos de estrutura dos modelos H0N3 (g) e H0S3 (h)

Fonte: Elaborado pelo próprio autor

A Figura 6.1 apresenta uma comparação entre os grafos de estrutura dos modelos D0N3 (a) e D0S3 (b). Nas figuras seguintes – Figura 6.2, Figura 6.3 e Figura 6.4 –, também há esse tipo de comparação. Por exemplo, no modelo D0S3, nota-se que o elemento 74 (setenta e quatro) se destaca. Tal destaque acontece em razão da concentração de conexões. No total, são 42 conexões simples e isso explica o tamanho de seu diâmetro. O elemento 74 retrata o componente *window3d* e é dentro desse componente que boa parte dos elementos 3D estão. Analisando os grafos de todas as figuras, o componente *window3d* se destaca, com exceção do grafo (f) da Figura 6.3. No modelo D1S3, grafo (f), existem 3 (três) componentes que se destacam mais do que o componente *window3d*. Esses três componentes são: *Eventos_Op01* (31), *Eventos_Op02* (32) e *Eventos_Op03* (33). Isso ocorre porque a estrutura lógica que atribui maior autonomia às entidades/agentes eleva o peso das conexões desses componentes e, conseqüentemente, esse fato aumenta os seus diâmetros.

Como já afirmado, o processo de adição dos elementos 3D consome uma quantidade considerável de tempo de modelagem, mas esse processo gerou modelos computacionais mais descritivos. A questão visual é importante, pois ajuda até mesmo no processo de verificação. No caso dessa tese, ao adicionar os elementos 3D, toda a estrutura lógica dos componentes ficou explícita no momento da simulação.

Para produzir os grafos, todas as fases do método iDAV, e suas etapas, foram aplicadas novamente. As conexões relacionadas com os elementos 3D são simples e possuem peso 1 (um). O critério de peso e os tipos de layout dos grafos foram mantidos. Sobre os parâmetros

estético, houve apenas a escolha da cor amarela para destacar os elementos 3D dentre os demais componentes de cada modelo. Ao analisar os grafos (b), (d), (f) e (h), é possível notar que os números que identificam os elementos 3D não são os mesmos. Isso se deve ao sequenciamento necessário para a construção das matrizes que servem de matéria-prima para a elaboração dos grafos.

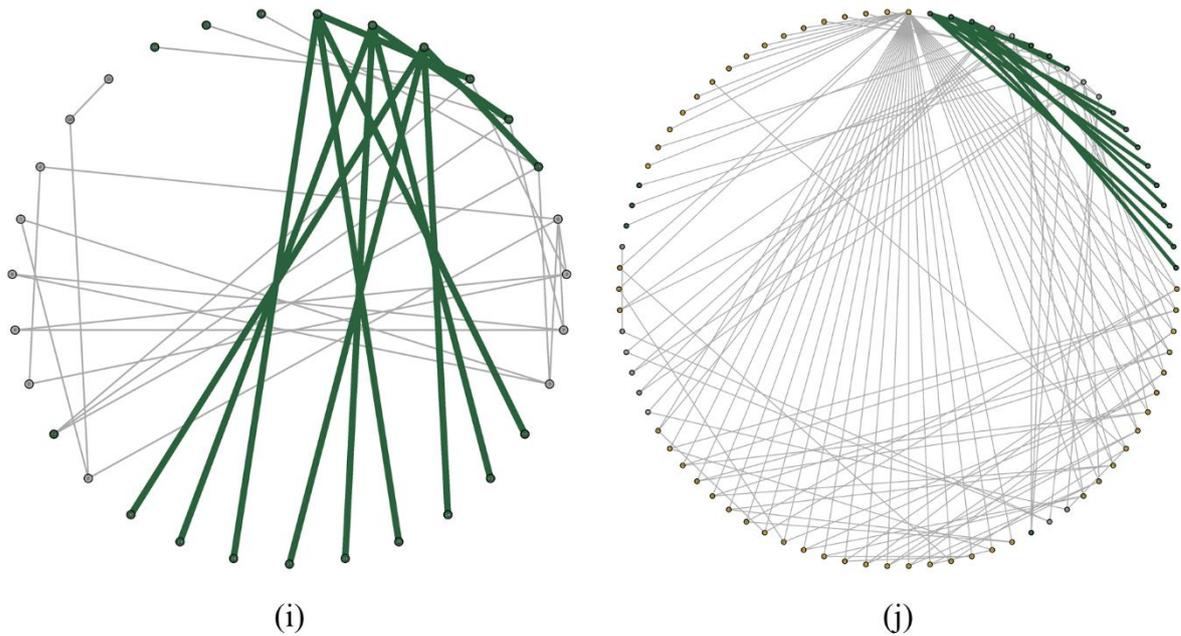


Figura 6.5 - Grafos de essência dos modelos D0N3 (i) e D0S3 (j)

Fonte: Elaborado pelo próprio autor

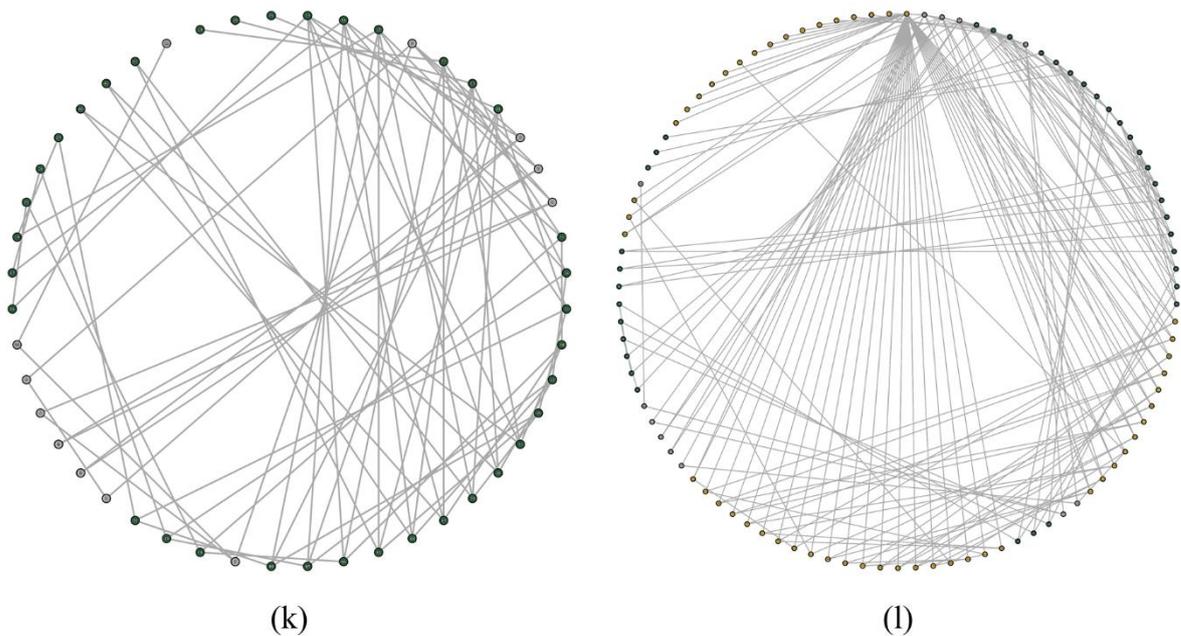
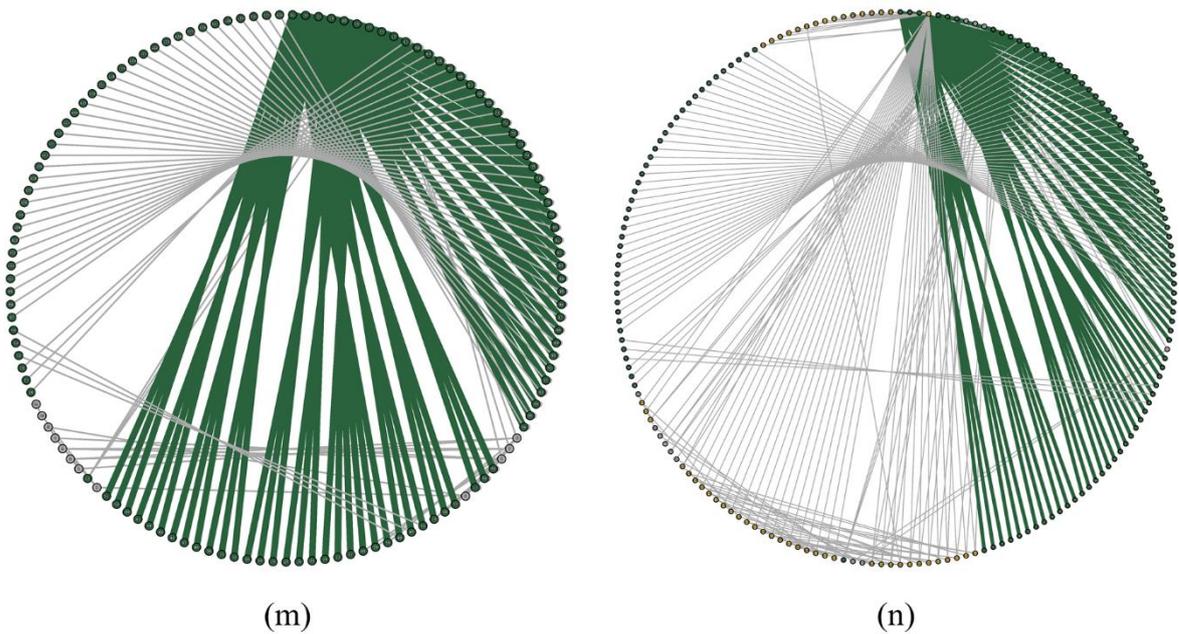
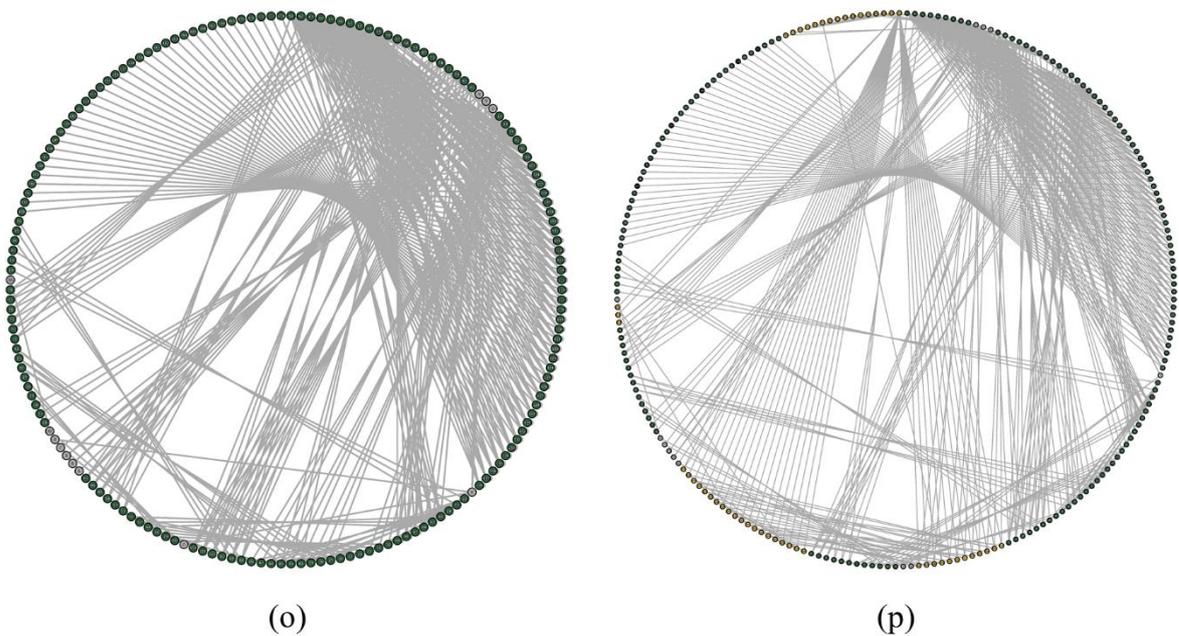


Figura 6.6 - Grafos de essência dos modelos H0N3 (k) e H0S3 (l)

Fonte: Elaborado pelo próprio autor



(m) (n)
 Figura 6.7 - Grafos de essência dos modelos D1N3 (m) e D1S3 (n)
 Fonte: Elaborado pelo próprio autor



(o) (p)
 Figura 6.8 - Grafos de essência dos modelos H1N3 (o) e H1S3 (p)
 Fonte: Elaborado pelo próprio autor

Houve um aumento significativo de componentes e conexões com a adição dos elementos 3D. Tal aumento é perceptível ao comparar os grafos de essência – Figura 6.5, Figura 6.6, Figura 6.7 e Figura 6.8. Nos grafos (j), (l), (n) e (p), percebe-se que os espaços vazios são menores. Apesar desse efeito, com a adição dos elementos 3D, a densidade dos grafos diminuiu. Conforme a literatura (GODEHARDT, 1988; KAUFMANN; WAGNER, 2001; TAMASSIA, 2014; WALLIS, 2007), a densidade de um determinado grafo consiste na razão entre a

quantidade de conexões presentes no grafo e o número máximo de conexões que esse grafo pode conter. Um grafo completo, no qual há todas as conexões possíveis, a densidade será sempre 1. A Tabela 6.1 mostra a diminuição mencionada.

Tabela 6.1 - Adição dos elementos 3D e a densidade dos grafos

Modelos computacionais	Densidade dos grafos
D0N3	0,0774
D0S3	0,0397
H0N3	0,0691
H0S3	0,0359
D1N3	0,0210
D1S3	0,0164
H1N3	0,0251
H1S3	0,0188

Fonte: Elaborado pelo próprio autor

Em termos percentuais, ao adicionar os elementos 3D, houve um aumento considerável de componentes e conexões. Na média, os componentes e conexões dos modelos computacionais aumentaram, respectivamente, 85% e 117%. A Tabela 6.2 contém o valor numérico dos componentes e conexões dos modelos computacionais tratados nessa seção.

Tabela 6.2 - Quantidade de componentes e conexões com a adição de elementos 3D

Modelos computacionais	Nº de componentes	Nº de conexões
D0N3	31	36
D0S3	82	132
H0N3	48	78
H0S3	99	174
D1N3	133	180
D1S3	184	276
H1N3	174	378
H1S3	225	474

Fonte: Elaborado pelo próprio autor

Como afirmado, os grafos de essência, que retratam os modelos computacionais que têm elementos 3D, possuem poucos espaços entre as conexões. A inclusão de elementos 3D produziu o aumento do número de componentes e conexões. Visualmente, os grafos de essência aparentam ser mais complexos do que aqueles em que não há elementos 3D. Tal observação também é possível de ser feita ao analisar os grafos de estrutura. Há um paradoxo, pois constata-se que o aumento do nível de detalhamento não gera um aumento da densidade dos grafos. Na verdade, o aumento do nível de detalhamento gera uma diminuição da densidade dos grafos. Essa premissa será abordada sob outra perspectiva na próxima seção na qual os postos de trabalhos serão triplicados.

6.2. Aumento dos postos de trabalho

Resgatando alguns dos princípios (SHANNON, 1975, p. 22, grifo do autor), é importante salientar que um bom modelo de simulação deve ser:

- *Adaptativo*, com um procedimento fácil para efetuar uma modificação ou atualização do modelo;
- *Evolutivo*, na medida em que o modelo deve iniciar de forma simples e se tornar mais complexos, em junção com o usuário.

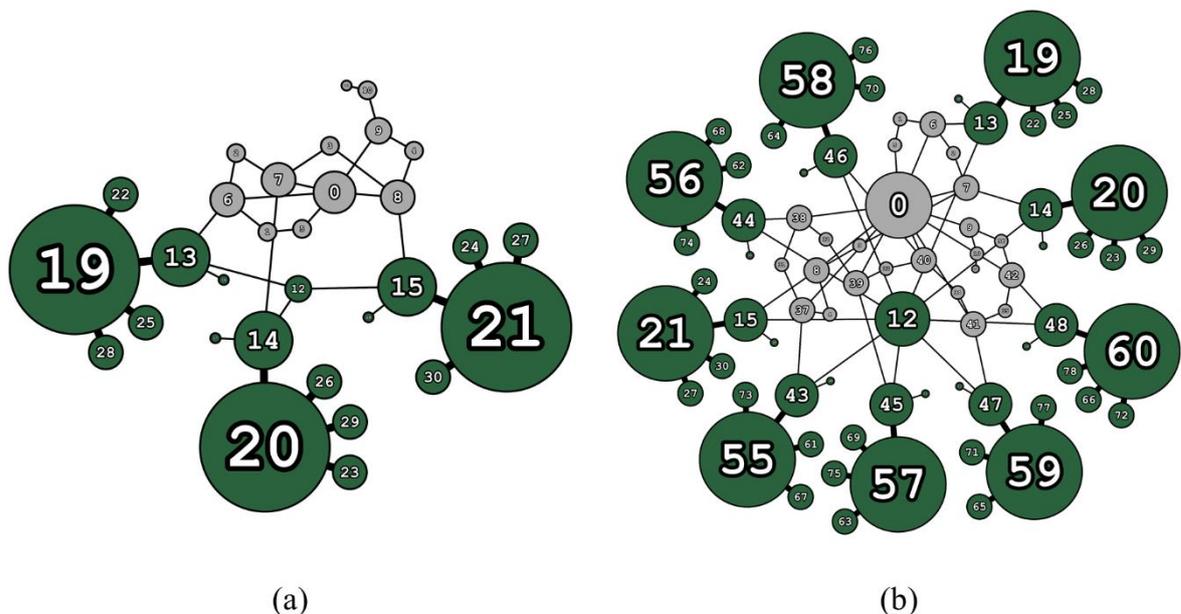


Figura 6.9 - Grafos de estrutura dos modelos D0N3 (a) e D0N9 (b)

Fonte: Elaborado pelo próprio autor

Nesta tese, os princípios supracitados foram colocados em prática. Em um primeiro momento, ambos os princípios se fazem presente quando o grau de autonomia das entidades/agentes foi

introduzido. Em um segundo momento, esses princípios foram novamente colocados em prática com a adição dos elementos 3D. Agora, esses princípios são utilizados mais uma vez como norte. Nessa seção, os postos de trabalhos foram triplicados e os efeitos visuais podem ser vistos nas Figuras 6.9, 6.10, 6.11 e 6.12.

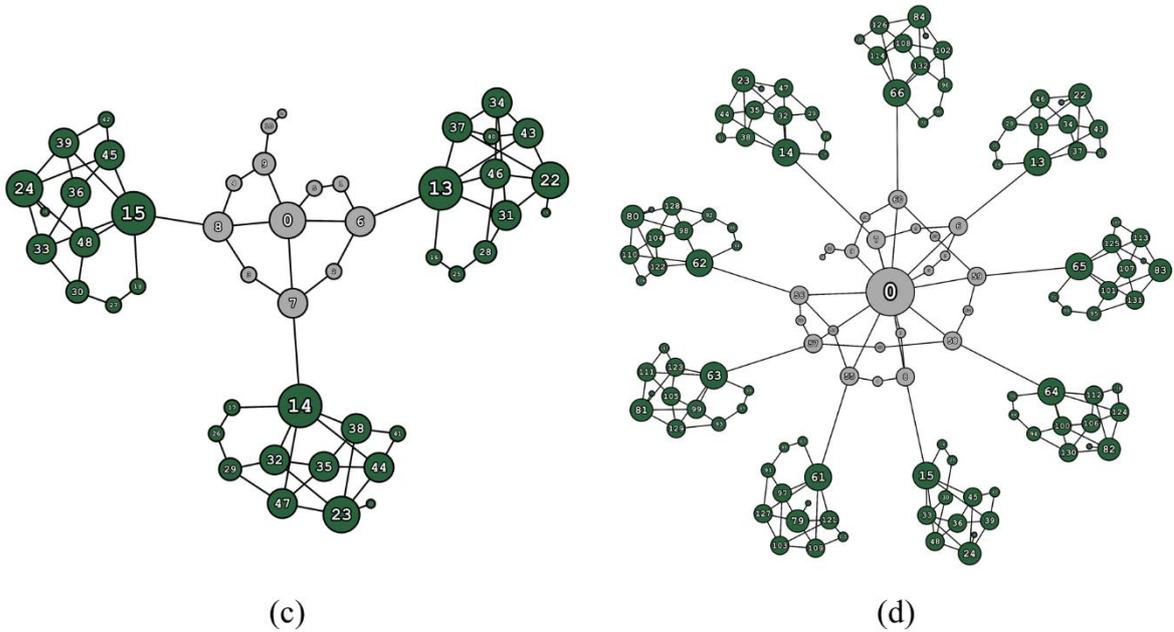


Figura 6.10 - Grafos de estrutura dos modelos H0N3 (c) e H0N9 (d)
Fonte: Elaborado pelo próprio autor

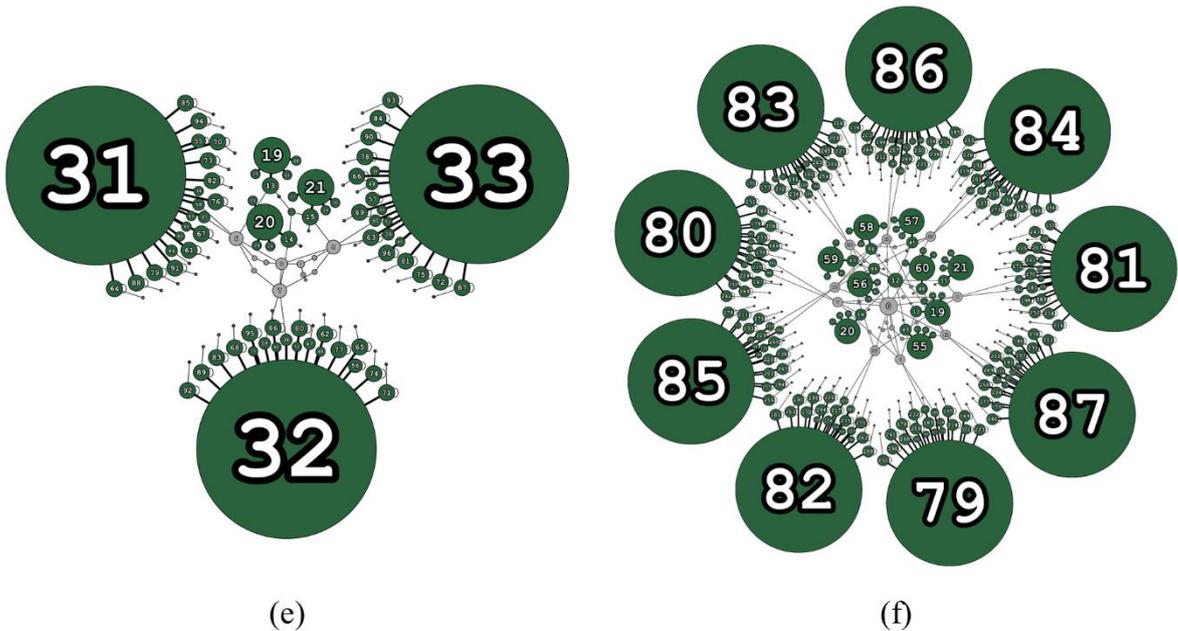


Figura 6.11 - Grafos de estrutura dos modelos D1N3 (e) e D1N9 (f)
Fonte: Elaborado pelo próprio autor

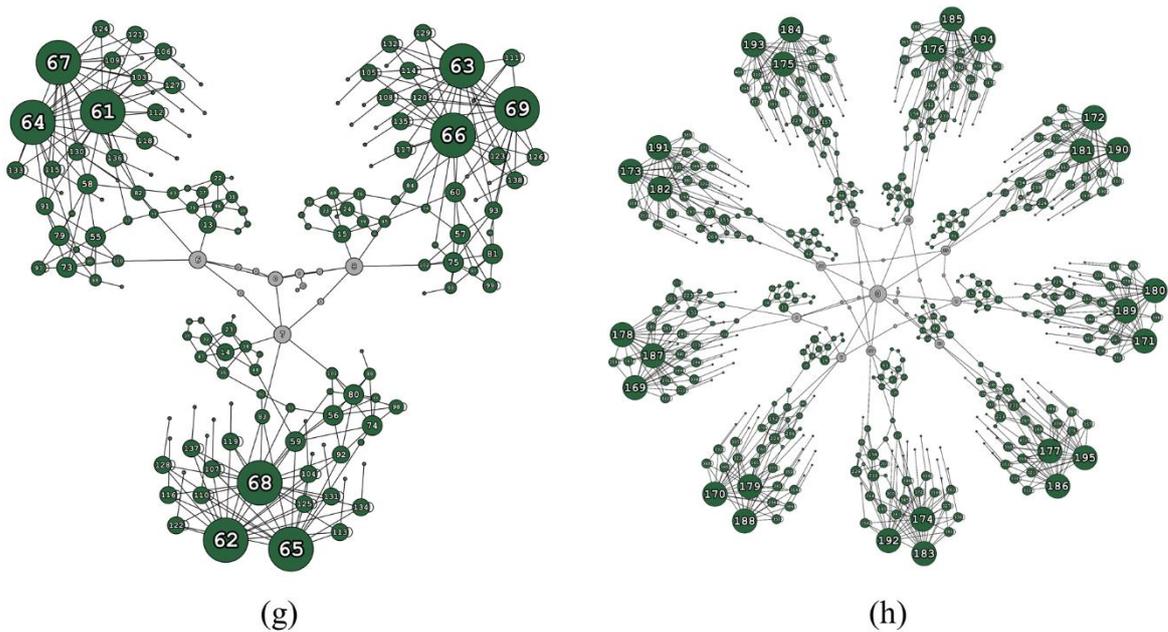


Figura 6.12 - Grafos de estrutura dos modelos H1N3 (g) e H1N9 (h)
 Fonte: Elaborado pelo próprio autor

Ao triplicar os postos de trabalho, passando de 3 (três) postos para 9 (nove), os grafos de estrutura ganham uma forma estrelada. Nos grafos de estrutura, os setores Fluxo de Processo (FP) e Comportamento da Entidade (CE) estão destacados, por essa ordem, em cinza e verde. Desse modo, o setor FP encontra-se no centro dos grafos de estrutura e o setor CE está nas extremidades, facilitando a análise sobre os componentes mais influentes dentro dos modelos computacionais.

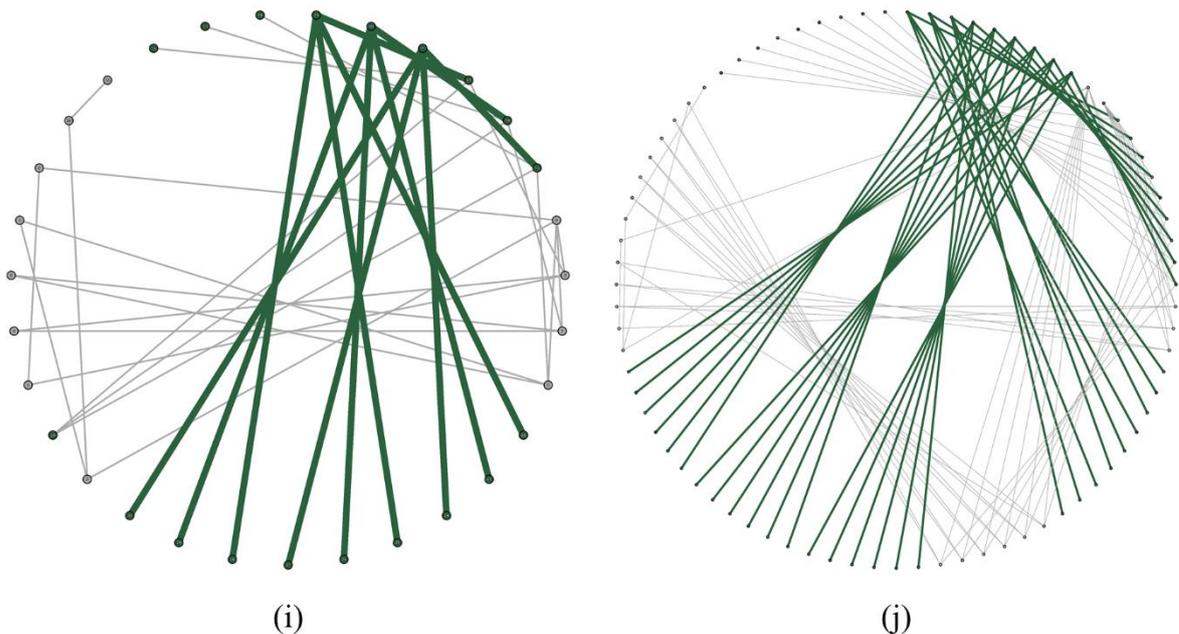


Figura 6.13 - Grafos de essência dos modelos D0N3 (i) e D0N9 (j)
 Fonte: Elaborado pelo próprio autor

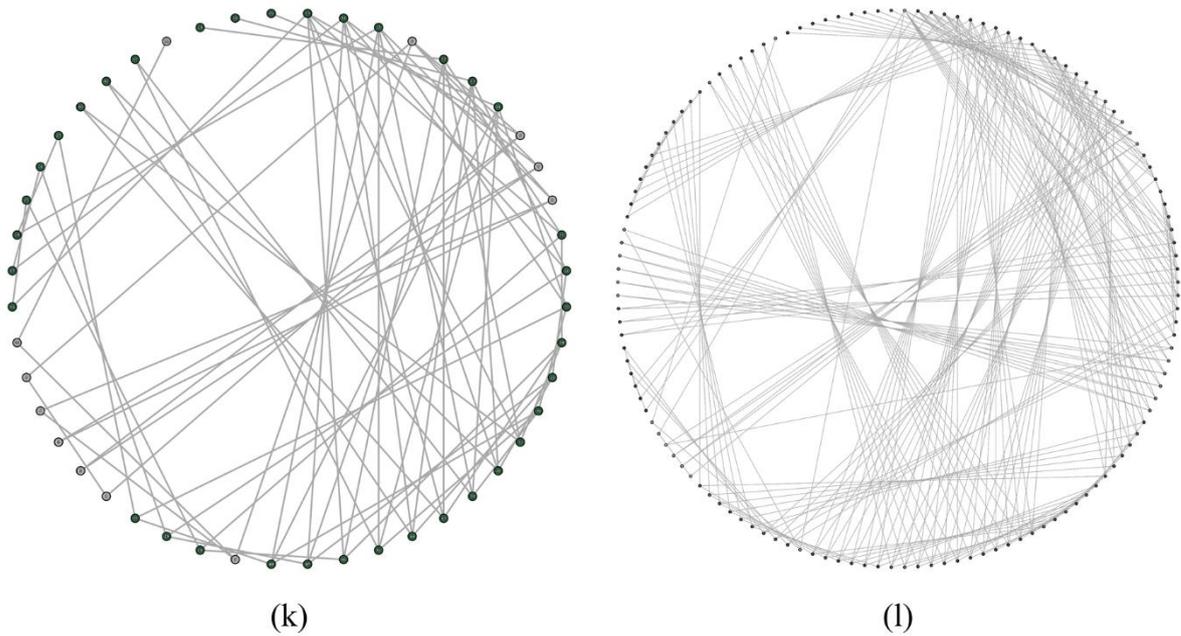


Figura 6.14 - Grafos de essência dos modelos H0N3 (k) e H0N9 (l)

Fonte: Elaborado pelo próprio autor

A produção dos grafos, tanto os de estrutura quanto os de essência, ocorreu por meio do método iDAV. Assim como foi relatado na seção anterior, todos os parâmetros foram mantidos. As Figuras 6.13 e 6.14, assim como as Figuras 6.15 e 6.16, mostram os grafos de essência dos modelos computacionais. Nota-se que esses grafos, cujo número de postos de trabalho é 9, ficaram mais complexos em razão da adição de mais componentes e, conseqüentemente, mais conexões.

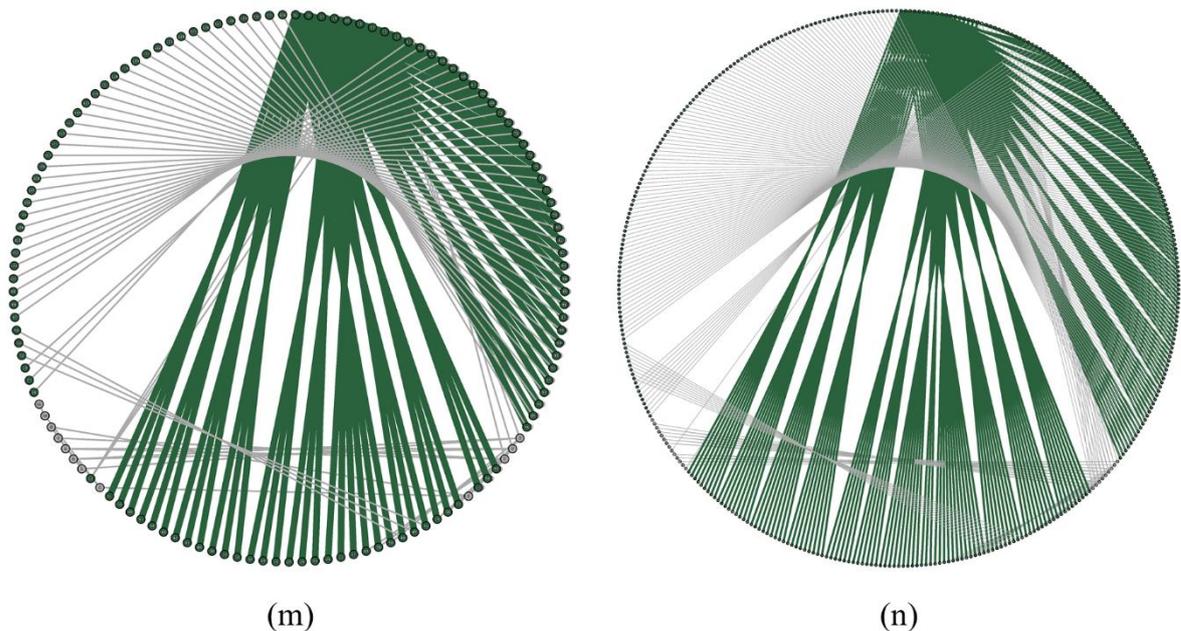


Figura 6.15 - Grafos de essência dos modelos D1N3 (m) e D1N9 (n)

Fonte: Elaborado pelo próprio autor

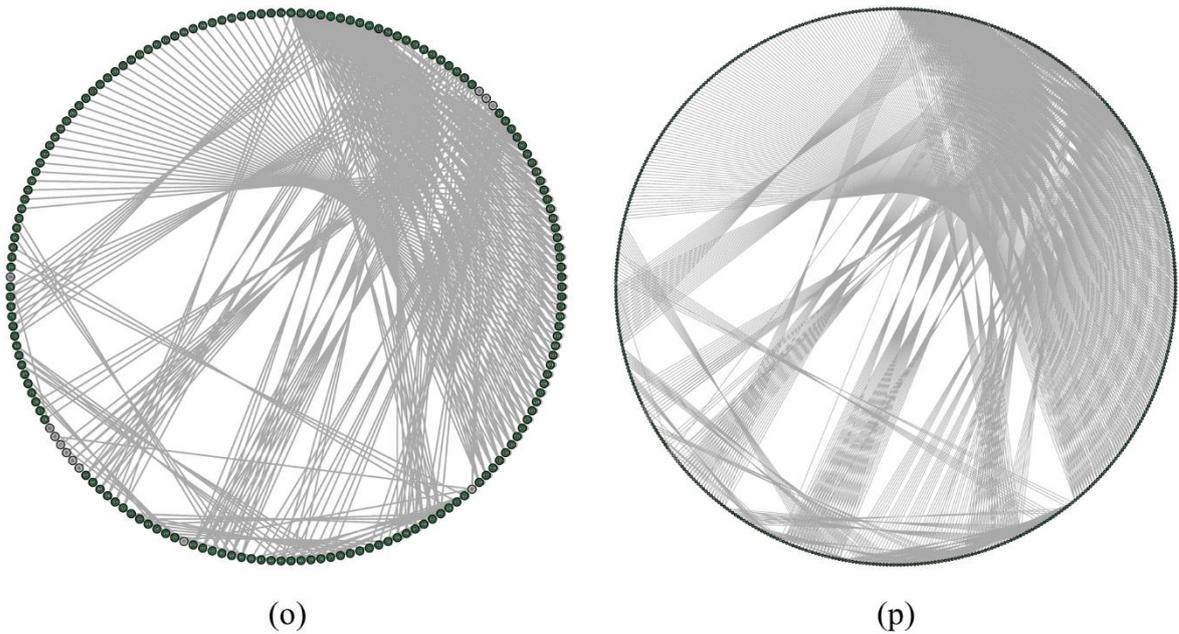


Figura 6.16 - Grafos de essência dos modelos H1N3 (o) e H1N9 (p)
 Fonte: Elaborado pelo próprio autor

Analisando os grafos dos modelos computacionais com 9 postos de trabalho, constata-se que densidade desses grafos é mais baixa. O paradoxo se confirma mais uma vez. Assim, o aumento do nível de detalhamento gera uma diminuição da densidade dos grafos. Na Tabela 6.3 estão as densidades de cada grafo.

Tabela 6.3 - Adição De novos postos de trabalho e a densidade dos grafos

Modelos computacionais	Densidade dos grafos
D0N3	0,0774
D0N9	0,0312
H0N3	0,0691
H0N9	0,0256
D1N3	0,0210
D1N9	0,0071
H1N3	0,0251
H1N9	0,0086

Fonte: Elaborado pelo próprio autor

De maneira geral, a densidade dos grafos é um indicador que está relacionado com o nível de detalhamento adotado nos modelos computacionais. Um bom modelo computacional deve, conforme os princípios abordados no início dessa seção, ser adaptativo e/ou evolutivo. Pode-

se aumentar a quantidade de componentes, como ocorreu no acréscimo de elementos 3D e de postos de trabalhos, ou aprofundar a quantidade de variáveis de um único componente, conforme aconteceu com a adição de maior grau de autonomia dos operadores. Independente de qual estratégia o especialista em simulação irá adotar, o fato é que a densidade dos grafos diminuirá. Principalmente se o nível de detalhamento de uma entidade/agente for aprofundado. A Figura 6.17 demonstra esse fato na forma de gráfico.

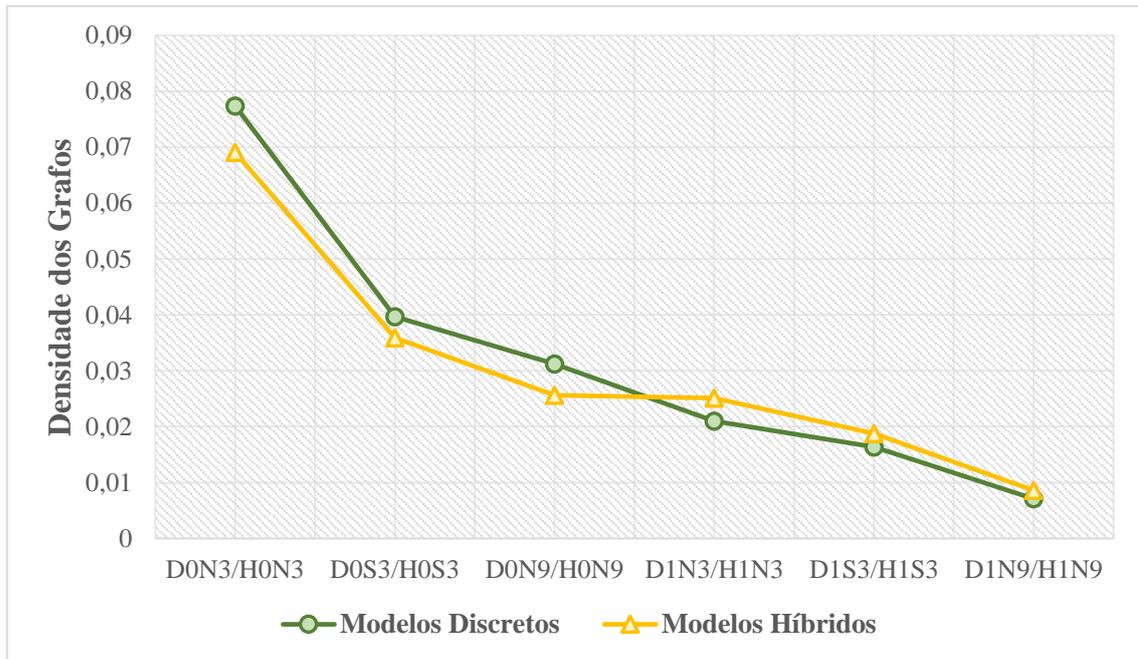


Figura 6.17 - Relação entre a densidade dos grafos e os modelos computacionais
Fonte: Elaborado pelo próprio autor

A queda da densidade dos grafos é uma tendência que pode ser comprovada, tanto nos modelos discretos quanto nos modelos híbridos, quando o escopo e o nível de detalhamento do modelo computacional são ampliados. A escolha do escopo e do nível de detalhe influenciam na complexidade do modelo computacional. Essa escolha gera impactos na densidade dos grafos.

7. AS CONCLUSÕES

7.1. Análises e Conclusões

Em razão do aumento da complexidade dos sistemas reais, a introdução de maior nível de detalhamento em modelos computacionais por meio da abordagem Simulação Híbrida (SH) é mais adequada? A complexidade dos modelos computacionais desenvolvidos pelas abordagens Simulação a Eventos Discretos (SED) e SH é a mesma?

Até o momento, os questionamentos supracitados no parágrafo anterior serviram como bússola para o desenvolvimento dessa tese. O nível de detalhamento abordado nesse trabalho científico repousa sobre o grau de autonomia das entidades/agentes presente nos modelos computacionais. A complexidade dos modelos computacionais foi aferida por meio da criação do método iDAV. Com o auxílio desse *método*, foram definidos parâmetros metodológicos para o processo de mensuração da complexidade de modelos computacionais. Essas ações viabilizaram a realização de comparações entre as abordagens SED e SH.

Por meio de um modelo conceitual hipotético, os modelos computacionais em que há aplicação das abordagens de SED e de SH foram comparados em termos estruturais. Essa comparação ocorreu no campo da complexidade. Entretanto, o que é algo complexo e como mensurar essa complexidade?

Um sistema pode ser classificado como complexo ou simples. O fato é que a ideia de complexidade pressupõe uma ordem dos elementos pertencentes a um sistema, porém essa ordem é desconhecida e, portanto, essa é a razão pela qual tudo aquilo que não se conhece com exatidão é definido como complexo. Se a ordem inexistente, então não se pode nomear o objeto de estudo como complexo, mas sim como caótico. Por outro lado, pode-se afirmar que a ideia de simplicidade pressupõe uma ordem dos elementos pertencentes a um sistema, mas essa ordem é conhecida e, dessa forma, dá-se o nome de simples para aquilo que se conhece e pode ser descrito com precisão.

Com essa ideia em mente, foi necessário evidenciar a ordem dentro daquilo que a princípio apresentava-se como complexo. Um modelo computacional, como todo sistema, é composto por componentes e conexões. Logo, para evidenciar uma ordem existente dentro dos modelos computacionais, foi necessário contabilizar quantos componentes e quais eram as conexões entre esses componentes dentro dos modelos. No processo de contagem desses elementos, notou-se que as conexões entre os componentes possuíam características diferentes e,

portanto, essas características deveriam ser salientadas. Com o intuito de exibir de modo claro a complexidade dos modelos computacionais, essa tese usou grafos.

Os grafos são ferramentas úteis tanto nos aspectos visuais quanto nos aspectos axiomáticos, *id est*, o Grau Médio Ponderado (GMP) dos modelos computacionais. Os valores do GMP já passam a ideia de como pode ser os aspectos visuais dos grafos de essência. O nome *grafo de essência* foi atribuído pelo próprio autor, pois esse tipo de grafo mostra de maneira clara a essência das conexões existentes entre os componentes dos modelos computacionais. Outro grafo utilizado nessa tese foi o *grafo de estrutura*. Esse nome faz referência a estrutura do modelo computacional em que se pode dividir em setores e analisá-los de modo isolado.

Nessa tese, como o foco de mudança está no nível de detalhamento do modelo, observa-se, a partir dos grafos de estrutura, que as entidades/agentes, ao possuir uma maior capacidade na tomada de decisão, influenciam no aumento da complexidade dos modelos computacionais. Conclui-se, portanto, que ao atribuir maior capacidade de tomada de decisão a um determinado componente de um modelo computacional, isso aumentará a complexidade do modelo. Em modelos de SED, o aumento da complexidade ocorre especificamente nas linhas de código. Nos modelos de SH, o aumento se manifesta no número de componentes e de conexões.

Nos modelos de SED, ao construir um modelo em que há um maior nível de tomada de decisão, o uso de estruturas lógicas como *if-else-if-else* e os métodos acabam se tornando elementos obrigatórios. Isso não é um problema para aqueles que possuem o conhecimento de programação. Por outro lado, por conta do uso de diagramas de estado, as linhas de código existentes nos modelos de SH são diluídas. Isso agiliza a programação, pois a estrutura lógica é construída através de diagramas de estados por meio de ações *drag-and-drop*. Assim, os comandos que fazem a lógica funcionar estão dentro das transições e estados.

Ao comparar as figuras e tabelas que exprimem as características de cada modelo computacional, constata-se que os modelos discretos são mais simples de serem desenvolvidos quando o nível de detalhe é menor. Por outro lado, quando o escopo e o nível de detalhamento são aumentados, os modelos híbridos são mais adequados. Esse fato ficou mais evidente ao analisar a densidade dos grafos. Invariavelmente, quando se aumenta o escopo e o nível de detalhamento, a densidade dos grafos diminuirá. Contudo, é maior a densidade dos grafos que retratam os modelos de SH com entidades/agentes mais autônomas. Isso indica que os modelos híbridos são mais adequados.

7.2. Sugestões para Pesquisas Futuras

A simulação é uma ferramenta em que é possível simular diferentes sistemas, sejam eles produtivos ou não. No caso dessa tese, um sistema de manufatura em linha serviu como base para efetuar análises sobre os modelos computacionais. Conforme constatado no *Capítulo 2*, há oportunidade de desenvolver pesquisas envolvendo o primeiro setor da economia. Somos um país no qual o agronegócio possui uma grande participação na geração de riqueza e a aplicação da simulação poderia gerar resultados interessantes.

Há na simulação diferentes tipos de abordagens, dentre eles estão a SED e a SH. Esta última abordagem é uma área recente da literatura científica, como foi demonstrado no *Capítulo 2*, e é por esse motivo existe campo para explorar, metodologicamente, a combinação e os efeitos desta combinação entre abordagens de simulação distintas.

Conforme a literatura científica, constatou-se que a escolha da abordagem de simulação é influenciada por experiências prévias do especialista. Esse é um dos motivos da não adoção de outras abordagens de simulação, como é o caso da abordagem SBA? A escolha se dá em razão da *expertise* do especialista ou essa escolha ocorre com base no sistema real que está sendo estudado? A discussão envolvendo a SED e a SBA suscita outra questão: qual abordagem de simulação é mais interessante, SED ou SH? Por serem abordagens distintas, o especialista decide combinar duas ou mais abordagens por necessidade? A partir dessas perguntas, uma pesquisa pode ser desenvolvida para compreender o processo de escolha das abordagens de simulação pelo especialista.

Resgatando o que foi afirmado por Robinson (2008), Figura 3.1 dessa tese, um modelo 100% preciso não é possível de ser alcançado devido a impossibilidade de capturar todas as variáveis presentes em um sistema real. Se um modelo computacional possui baixo escopo e nível de detalhamento, isso irá torná-lo menos preciso. O mesmo ocorre quando o escopo e nível de detalhamento é grande demais. Nesta tese, a representação de um modelo computacional por meio de grafos, auxiliou na análise visual de seus componentes e conexões. Ainda em termos visuais, um grafo pode aparentar ser mais complexo do que outro, e isso é um indicativo de que há diferenças entre as densidades desses grafos. Será que há relação entre a densidade desses grafos e a Figura 3.1 de Robinson (2008)? Será que a complexidade dos modelos computacionais é inversamente proporcional ao valor da densidade do grafo, *id est*, a densidade do grafo diminui à medida em que um modelo de

simulação se torna mais complexo? Mais estudos deverão ser feitos para transformação essas perguntas em afirmações.

Aparentemente, a combinação ou não das abordagens de simulação gera irrelevante impacto quanto ao valor da densidade dos grafos. Apesar disso, nessa tese, nos modelos em que as entidades/agentes são mais autônomas, há aumento de conexões mais elaboradas entre os componentes. Basta saber se este padrão se mantém para outros tipos de sistema reais simulados, como hospitais, restaurantes, eventos, fazendas, etc.

Por fim, os pesquisadores poderão aplicar o método iDAV em modelos discretos e simulação híbrida que retratam outros segmentos da área de manufatura como, por exemplo: sistemas de movimentação, sistemas de armazenagem, células automatizadas, programação da produção, análise de estoques e análise de *kanban*. O método iDAV também pode ser usado para abrir um novo campo de estudo, que trataria sobre a qualidade dos modelos computacionais. Isso seria interessante, porque, por exemplo, a partir do método iDAV, o especialista poderia verificar se seu modelo computacional possui componentes supérfluos.

AS REFERÊNCIAS BIBLIOGRÁFICAS

ABEL, K. C.; FAUST, K. M. Modeling complex human systems: An adaptable framework of urban food deserts. **Sustainable Cities and Society**, v.52, pp.1-13, 2020.

ALEXANDRE, M.; LIMA, G. T. Combining monetary policy and prudential regulation: an agent-based modeling approach. **Journal of Economic Interaction and Coordination**, v.15, pp.385-411, 2020.

ARENALES, M.; ARMENTANO, V.; MORABITO, R.; YANASSE, H. **Pesquisa Operacional**. Rio de Janeiro: Campus/Elsevier 2007.

AZEVEDO, R. V.; MOURA, M. C.; LINS, I. D.; DROGUETT, E. L. A multi-objective approach for solving a replacement policy problem for equipment subject to imperfect repairs. **Applied Mathematical Modelling**, v.86, pp.1-19, 2020.

BAINES, T.; MASON, S.; SIEBERS, P. O.; LADBROOK, J. Humans: the missing link in manufacturing simulation? **Simulation Modelling Practice and Theory**, v.12, pp.515-526, 2004.

BARBIERATO, E.; GRIBAUDO, M.; SERAZZI, G. Multi-formalism Models for Performance Engineering. **Future Internet**, v.12, n.50, pp.1-26, 2020.

BARBIERI, J. P.; PINHO, A. F.; MONTEVECHI, J. A. B. Autonomous Entities: a hybrid model and its effects. **International Journal of Simulation Modelling**, v.22, n.2, pp. 187-198, 2023.

BARNES, R. M. **Estudos de movimentos e de tempos: projetos e medidas do trabalho**. 6. ed. São Paulo: Edgard Blücher, 1977.

BERTALANFFY, L. **General Systems Theory**. New York: Braziller, 1968.

BERTRAND, J. W.; FRANSOO, J. C. Modelling and Simulation: operations management research methodologies using quantitative modeling. **International Journal of Operations & Production Management**, v.22, n.2, pp.241-264, 2002.

BERTUGLIA, C. S.; VAIO, F. **Nonlinearity, Chaos, and Complexity: The Dynamics of Natural and Social Systems**. 1. ed. New York: Oxford University Press, 2005.

BHATTACHERJEE, A. **Social Science Research: principles, methods, and practices**. 2. ed. Tampa: CreateSpace Independent Publishing Platform, 2012.

BRAILSFORD, S. C.; HARPER, P. R.; SYKES, J. Incorporating human behaviour in simulation models of screening for breast cancer. **European Journal of Operational Research**, v.219, pp.491-507, 2012.

BRAILSFORD, S. C. Modeling Human Behavior - an (id)entity crisis? In: Winter Simulation Conference, **Proceedings...** Savannah, GA, USA, 2014a.

BRAILSFORD, S. C. Discrete-event simulation is alive and kicking! **Journal of Simulation**, v.8, pp.1-8, 2014b.

BRAILSFORD, S. C.; EL DABI, T.; KUNC, M.; MUSTAFEE, N.; OSORIO, A. F. Hybrid simulation modelling in operational research: a state-of-the-art review. **European Journal of Operational Research**, v.278, pp.721-737, 2019.

BROOKS, R. J.; TOBIAS, A. M. Choosing the Best Model: Level of Detail, Complexity, and Model Performance. **Mathematical and Computer Modelling**, v.24, n.4, pp.1-14, 1996.

- CHEN, S.; LU, W.; OLOFSSON, T.; DEHGHANIMOHAMMADABADI, M.; EMBORG, M.; NILIMAA, J.; WANG, Y.; FENG, K. Concrete Construction: How to Explore Environmental and Economic Sustainability in Cold Climates. **Sustainability**, v.12, pp.1-23, 2020.
- CHUDZINSKA, M.; DUPONT, Y. L.; NABE-NIELSEN, J.; MAIA, K. P.; HENRIKSEN, M. V.; RASMUSSEN, C.; KISSLING, W. D.; HAGEN, M.; TRØJELSGAARD, K. Combining the strengths of agent-based modelling and network statistics to understand animal movement and interactions with resources: example from within-patch foraging decisions of bumblebees. **Ecological Modelling**, v.430, pp.1-12, 2020.
- CHWIF, L.; MEDINA, A. C. **Modelagem e Simulação de eventos discretos: teoria & aplicações**. 4. ed. Rio de Janeiro: Elsevier, 2015.
- CICIRELLI, F.; FURFARO, A.; NIGRO, L. Modelling and simulation of complex manufacturing systems using statechart-based actors. **Simulation Modelling Practice and Theory**, v.19, pp.685-703, 2011.
- DELCEA, C.; COTFAS, L.; BRADEA, I.; BOLOS, M.; FERRUZZI, G. Investigating the Exits' Symmetry Impact on the Evacuation Process of Classrooms and Lecture Halls: An Agent-Based Modeling Approach. **Symmetry**, v.12, n.627, pp.1-25, 2020.
- DI BATTISTA, G.; EADES, P.; TAMASSIA, R.; TOLLIS, I. G. Algorithms for drawing graphs: an annotated bibliography. **Computational Geometry**, v.4, pp.235-282, 1994.
- ELDABI, T.; BELL, D.; TAKO, A. A.; TOLK, A. Tutorial on means of hybrid simulation. In: Winter Simulation Conference, **Proceedings...** National Harbor, MD, USA, 2019.
- FAKHIMI, M.; ANAGNOSTOU, A.; STERGIOULAS, L.; TAYLOR, S. J. E. A Hybrid Agent-Based and Discrete Event Simulation Approach for Sustainable Strategic Planning and Simulation Analytics. In: Winter Simulation Conference, **Proceedings...** Savannah, GA, USA, 2014.
- FANI, V.; PIROLA, F.; BINDI, B.; BANDINELLI, R.; PEZZOTTA, G. Design product-service systems by using a Hybrid Approach: The fashion renting business model. **Sustainability**, v.14, pp.1-16, 2022.
- FERNANDEZ-MENA, H.; GAUDOU, B.; PELLERIN, S.; MACDONALD, G. K.; NESME, T. Flows in Agro-food Networks (FAN): An agent-based model to simulate local agricultural material flows. **Agricultural Systems**, v.180, pp.1-17, 2020.
- FIORETTI, G. Agent-Based Simulation Models in Organization Science. **Organizational Research Methods**, v.16, n.2, pp.227-242, 2012.
- FORREST, J. Y. **General Systems Theory: Foundation, Intuition and Applications in Business Decision Making**. Switzerland: Springer, 2018.
- FRANCESCHINI, F.; MAISANO, D.; MASTROGIACOMO, L. Scientific journal publishers and omitted citations in bibliometric databases: Any relationship? **Journal of Informetrics**, v.8, n.3, pp.751-765, 2014.
- GIANNASI, F.; LOVETT, P.; GODWIN, A. N. Enhancing confidence in discrete event simulations. **Computers in Industry**, v.44, pp.141-157, 2001.
- GODEHARDT, E. **Graphs as structural models: the application of graphs and multigraphs in cluster analysis**. 4 ed. Wiesbaden: Vieweg, 1988.
- GOLDSMAN, D.; NANCE, R. E.; WILSON, J. R. A brief history of simulation revisited. In: Winter Simulation Conference, **Proceedings...** Baltimore, MD, USA, 2010.

- GORDON, G. **System Simulation**. New Jersey: Prentice-Hall, 1969.
- GUL, M.; GUNERI, A. F.; GUNAL, M. M. Emergency department network under disaster conditions: The case of possible major Istanbul earthquake. **Journal of the Operational Research Society**, v.71, n.5, pp.733-747, 2020.
- GUNAL, M. M. Simulation and the Fourth Industrial Revolution. In: GUNAL, M. M. (Ed). **Simulation for Industry 4.0: Past, Present, and Future**. Switzerland: Springer, 2019.
- HARRISON, J. R.; LIN, Z.; CARROLL, G. R.; CARLEY, K. M. Simulation modeling in organizational and management research. **Academy of Management Review**, v.32, n.4, pp.1229-1245, 2007.
- HASIĆ, F.; VANTHIENEN, J. Complexity metrics for DMN decision models. **Computer Standards & Interfaces**, v.65, pp.15-37, 2019.
- HERMAN, I.; MELANÇON, G.; MARSHALL, M. S. Graph Visualization and Navigation in Information Visualization: A Survey. **IEEE Transactions on Visualization and Computer Graphics**, v.6, n.1, pp.24-43, 2000.
- HWANG, I. An Agent-Based Model of Firm Size Distribution and Collaborative Innovation. **Journal of Artificial Societies and Social Simulation**, v.23(1), n.9, pp.1-21, 2020.
- INGALLS, R. G. Introduction to simulation. In: Winter Simulation Conference, **Proceedings...** Miami, FL, USA, 2008.
- JACOMY, M.; VENTURINI, T.; HEYMANN, S. BASTIAN, M. ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software. **PLOS ONE**, v.9, n.6, pp.1-12, 2014.
- JAHANGIRIAN, M.; ELDABI, T.; NASEER, A.; STERGIOULAS, L. K.; YOUNG, T. Simulation in manufacturing and business: A review. **European Journal of Operational Research**, v.203, pp.1-13, 2010.
- KALB, G. E. Counting lines of code, confusions, conclusions, and recommendations. In: Briefing to the 3rd Annual REVIC User's Group Conference, **Proceedings...**, 1990.
- KAMBLI, A.; SINHA, A. A.; SRINIVAS, S. Improving campus dining operations using capacity and queue management: A simulation-based case study. **Journal of Hospitality and Tourism Management**, v.43, pp.62-70, 2020.
- KASAIE, P.; KELTON, W. D. Guidelines for Design and Analysis in Agent-Based Simulation Studies. In: Winter Simulation Conference, **Proceedings...** Huntington Beach, CA, USA, 2015.
- KAUFMANN, M.; WAGNER, D. **Drawing Graphs: Methods and Models**. Berlin: Springer, 2001.
- KOREN, Y. Drawing Graphs by Eigenvectors: Theory and Practice. **Computers and Mathematics with Applications**, v.49, pp.1867-1888, 2005.
- LAW, A. M. Simulation of manufacturing systems. In: Winter Simulation Conference, **Proceedings...** San Diego, CA, USA, 1988.
- LAW, A. M. **Simulation Modeling & Analysis**. 4 ed. New York: McGraw-Hill, 2007.
- LAW, A. M. **Simulation Modeling & Analysis**. 5 ed. New York: McGraw-Hill, 2015.
- LEAL, F. **Análise do efeito interativo de falhas em processos de manufatura através de projeto de experimentos simulados**. 2008. 237 p. Tese (Doutorado em Engenharia

Mecânica) – Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, SP, 2008.

LITTLE, J. D. C. Models and Managers: the concept of a decision calculus. **Management Science**, v.16, n.8, pp.466-485, 1970.

MACAL, C. M.; NORTH, M. J. Tutorial on Agent-Based Modeling and Simulation. In: Winter Simulation Conference, **Proceedings...** Orlando, FL, USA, 2005.

MACAL, C. M.; NORTH, M. J. Agent-Based Modeling and Simulation. In: Winter Simulation Conference, **Proceedings...** Austin, TX, USA, 2009.

MACAL, C. M.; NORTH, M. J.; SAMUELSON, D. A. Elements of Agent-Based Simulation. In: GASS, S. I.; FU, M. C. (Ed). **Encyclopedia of Operations Research and Management Science**. 3. ed. New York: Springer, 2013. Cap.Agent-Based Simulation, pp.8-16.

MACAL, C.; NORTH, M. Introductory Tutorial: Agent-Based Modeling and Simulation. In: Winter Simulation Conference, **Proceedings...** Savannah, GA, USA, 2014.

MACAL, C. M. Everything you need to know about agent-based modelling and simulation. **Journal of Simulation**, v.10, n.02, pp.144-156, 2016.

MARSHALL, D. A.; TAGIMACRUZ, T.; CEPOIU-MARTIN, M.; ROBERT, J.; RING, B.; BURSTON, M.; HIGGINS, S.; HESS, M.; WHITE, J. A Simulation Modelling Study of Referral Distribution Policies in a Centralized Intake System for Surgical Consultation. **Journal of Medical Systems**, v.47, n.4, 2023.

MARTINS, R. A. Abordagens Quantitativa e Qualitativa. In: MIGUEL, P. A. C. (Org). **Metodologia de Pesquisa em Engenharia de Produção e Gestão de Operações**. 2. ed. Rio de Janeiro: Elsevier, 2012. Cap.3, pp.47-63.

MAYAKA, J.; JUNG, J. C. Complexity reduction of the Engineered Safety Features Component Control System. **Nuclear Engineering and Design**, v.331, pp.194-203, 2018.

MCCABE, T. J. A complexity measure. **IEEE Transactions on Software Engineering**, v.2, n.4, pp.308-320, 1976.

MITROFF, I. I.; BETZ, F.; PONDY, L. R.; SAGASTI, F. On Managing Science in the Systems Age: Two Schemas for the Study of Science as a Whole Systems Phenomenon. **Interfaces**, v.4, n.3, pp.46-58, 1974.

MORABITO NETO, R.; PUREZA, V. Abordagens Quantitativa e Qualitativa. In: MIGUEL, P. A. C. (Org). **Metodologia de Pesquisa em Engenharia de Produção e Gestão de Operações**. 2. ed. Rio de Janeiro: Elsevier, 2012. Cap.8, pp.169-198.

MORTAZAVI, A.; KHAMSEH, A. A.; AZIMI, P. Designing of an intelligent self-adaptive model for supply chain ordering management system. **Engineering Applications of Artificial Intelligence**, v.37, pp.207-220, 2015.

MUSTAFEE, N.; POWELL, J. H. From Hybrid Simulation to Hybrid Systems Modelling. In: Winter Simulation Conference, **Proceedings...** Gothenburg, Sweden, 2018.

MYKONIATIS, K.; ANGELOPOULOU, A. A modeling framework for the application of multi-paradigm simulation methods. **Simulation: Transactions of the Society for Modeling and Simulation International**, v.96, n.1, pp.55-73, 2020.

NAGADI, K; RABELO, L.; BASINGAB, M.; SARMIENTO, A. T.; JONES, A.; RAHAL, A. A hybrid simulation-based assessment framework of smart manufacturing systems.

International Journal of Computer Integrated Manufacturing, v.31, n.2, pp.115-128, 2018.

NANCE, R. E.; SARGENT, R. G. Perspectives on the evolution of simulation. **Operations Research**, v.50, n.1, pp.161-172, 2002.

NEGAHBAN, A.; SMITH, J. S. Simulation for manufacturing system design and operation: literature review and analysis. **Journal of Manufacturing Systems**, v.33, pp.241-261, 2014.

OLIVEIRA, M. L. M.; MONTEVECHI, J. A. B.; PINHO, A. F.; MIRANDA, R. C. Using Hybrid Simulation to Represent the Human Factor in Production Systems. **International Journal of Simulation Modelling**, v.16, n.2, pp.263-274, 2017.

OZALP, M. K.; MILLER, L. A.; DOMBROWSKI, T.; BRAYE, M.; DIX, T.; PONGRACZ, L.; HOWELL, R.; KLOTSKA, D.; PASOUR, V.; STRICKLAND, C. Experiments and Agent Based Models of Zooplankton Movement within Complex Flow Environments. **Biomimetics**, v.5, n.1, pp.1-24, 2020.

PERGHER, I.; FREJ, E. A.; ROSELLI, L. R. P.; DE ALMEIDA, A. T. Integrating simulation and FITradeoff method for scheduling rules selection in job-shop production systems. **International Journal of Production Economics**, v.227, pp.1-10, 2020.

POLANČIČ, G.; CEGNAR, B. Complexity metrics for process models – A systematic literature review. **Computer Standards & Interfaces**, v.51, pp.104-117, 2017.

POPOVICS, G.; MONOSTORI, L. An approach to determine simulation model complexity. **Procedia CIRP**, v.52, pp.257–261, 2016.

POSSIK, J.; ASGARY, A.; SOLIS, A. O.; ZACHAREWICZ, G.; SHAFIEE, M. A.; NAJAFABADI, M. M.; NADRI, N.; GUIMARAES, A.; IRANFAR, H.; MA, P.; LEE, C. M.; TOFIGHI, M. An Agent-Based Modeling and Virtual Reality Application Using Distributed Simulation: Case of a COVID-19 Intensive Care Unit. **IEEE Transactions on Engineering Management**, v.70, pp.2931-2943, 2023.

PRODANOV, C. C.; FREITAS, E. C. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico**. 2 ed. Novo Hamburgo: Feevale, 2013.

RAEDT, H. D.; JATTANA, M. S.; WILLSCH, D.; WILLSCH, M.; JIN, F.; MICHIELSEN, K. Discrete-Event Simulation of an Extended Einstein-Podolsky-Rosen-Bohm Experiment. **Frontiers in Physics**, v.8, pp.1-13, 2020.

ROBINSON, S. Conceptual modelling for simulation Part I: definition and requirements. **Journal of the Operational Research Society**, v.59, pp.278-290, 2008.

ROBINSON, S. Conceptual modeling for simulation. In: Winter Simulation Conference, **Proceedings...** Washington, DC, USA, 2013.

RODRIGUES, R. P.; PINHO, A. F.; SENA, D. C. Application of Hybrid Simulation in production scheduling in job shop systems. **Simulation: Transactions of the Society for Modeling and Simulation International**, pp.1-16, 2019.

SARGENT, R. G. Verification and validation of simulation models. **Journal of Simulation**, v.7, pp.12-24, 2013.

SARGENT, R. G. Verifying and validating simulation models. In: Winter Simulation Conference, **Proceedings...** Savannah, GA, USA, 2014.

- SARJOUGHIAN, H. S. Restraining complexity and scale traits for component-based simulation models. In: Winter Simulation Conference, **Proceedings...** Las Vegas, NV, USA, 2017.
- SCHEIDEGGER, A. P. G.; PEREIRA, T. F.; OLIVEIRA, M. L. M.; BANERJEE, A.; MONTEVECHI, J. A. B. An introductory guide for hybrid simulation modelers on the primary simulation methods in industrial engineering identified through a systematic review of the literature. **Computers & Industrial Engineering**, v.124, pp.474-492, 2018.
- SCHÖNEMANN, M.; HERRMANN, C.; GRESCHKE, P.; THIEDE, S. Simulation of matrix-structure manufacturing systems. **Journal of Manufacturing Systems**, v.37, pp.104-112, 2015.
- SENA, D. C. **Política de Gestão de Estoques Utilizando Aprendizado por Reforço e Simulação Híbrida em uma Farmácia Hospitalar**. 2017. 156 p. Tese (Doutorado em Engenharia de Produção) – Universidade Federal de Itajubá, Itajubá, MG, 2017.
- SENA, D. C.; SILVA, E. M. M.; COSTA, A. P. R.; MONTEVECHI, J. A. B.; PINHO, A. F.; MIRANDA, R. C. Dynamic Allocation of Additional Human Resources Using Hybrid Simulation. **International Journal of Simulation Modelling**, v.16, n.1, pp.84-95, 2017.
- SHANNON, R. E. **Systems Simulation: the art and science**. Englewood Cliffs: Prentice-Hall, 1975.
- SHAO, J.; WANG, Y. A new measure of software complexity based on cognitive weights. **IEEE Canadian Journal of Electrical and Computer Engineering**, v.28, n.2, pp.69-74, 2003.
- SIEBERS, P. O.; MACAL, C. M.; GARNETT, J.; BUXTON, D.; PIDD, M. Discrete-event simulation is dead, long live agent-based simulation! **Journal of Simulation**, v.4, n.3, pp.204-210, 2010.
- SIMON, H. A. The Architecture of Complexity. **Proceedings of the American Philosophical Society**, v.106, n.6, pp.467-482, 1962.
- STURROCK, D. T. Tutorial: Tips for Successful Practice of Simulation. In: Winter Simulation Conference, **Proceedings...** Savannah, GA, USA, 2014.
- TAKO, A. A.; FISHWICK, P.; ELDABI, T.; KREJCI, C. C.; KUNC, M. Panel - Towards Conceptual Modeling for Hybrid Simulation: Setting the Scene. In: Winter Simulation Conference, **Proceedings...** National Harbor, MD, USA, 2019.
- TAMASSIA, R. **Handbook of Graph Drawing and Visualization**. Boca Raton: CRC Press, 2014.
- TOLL, W. V.; PETTRÉ, J. Synchronizing navigation algorithms for crowd simulation via topological strategies. **Computers & Graphics**, v.89, pp.24-37, 2020.
- VIANA, J.; BRAILSFORD, S. C.; HARINDRA, V.; HARPER, P. R. Combining discrete-event simulation and system dynamics in a healthcare setting: A composite model for Chlamydia infection. **European Journal of Operational Research**, v.237, pp.196-206, 2014.
- VINOGRADOV, E.; LEICK, B.; KIVEDAL, B. K. An agent-based modelling approach to housing market regulations and Airbnb-induced tourism. **Tourism Management**, v.77, pp.1-9, 2020.
- VIZZARI, G.; CROCIANI, L.; BANDINI, S. An agent-based model for plausible wayfinding in pedestrian simulation. **Engineering Applications of Artificial Intelligence**, v.87, pp.1-13, 2020.

WALLIMAN, N. **Research Methods: the basics**. New York: Routledge, 2011.

WALLIS, W. D. **A beginner's guide to graph theory**. 2 ed. Boston: Birkhäuser, 2007.

YÜCESAN, E.; SCHRUBEN, L. Complexity of Simulation Models: A Graph Theoretic Approach. **INFORMS Journal on Computing**, v.10, n.1, pp.94-106, 1998.

ZAKIKHANI, K.; NASIRI, F.; ZAYED, T. Availability-based reliability-centered maintenance planning for gas transmission pipelines. **International Journal of Pressure Vessels and Piping**, v.183, pp.1-9, 2020.

ZEIGLER B. P.; PRAEHOFER, H.; KIM, T. G. **Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems**. 2. ed. San Diego: Academic Press, 2000.