

COMPARAÇÃO DA COMPLEXIDADE ENTRE MODELOS DE SIMULAÇÃO A EVENTOS DISCRETOS E HÍBRIDOS NO RAMO DA MANUFATURA

Tiago Dela Savia

tiagodsavia@outlook.com

Alexandre Ferreira de Pinho

pinho@unifei.edu.br

João Paulo Barbieri

j.p.barbieri@hotmail.com

Matheus Murno Bernardo

matheusmurno@gmail.com

Instituto de Produção e Gestão
Universidade Federal de Itajubá
Av. B P S, 1303 - Pinheirinho, Itajubá - MG

RESUMO

A simulação é uma ferramenta que permite facilitar o estudo de um sistema real ao representá-lo em um ambiente computacional. Entre as abordagens que podem ser usadas para se representar um modelo, podemos destacar a Simulação a Eventos Discretos (SED) e a Simulação Baseada em Agentes (SBA), que podem ser combinadas gerando a Simulação Híbrida (SH). Este estudo analisa dois modelos de simulação que representam uma situação hipotética de manufatura *flow-shop*, sendo um modelo construído a partir da SED e outro a partir da SH, para avaliar a complexidade dos dois e verificar qual das duas abordagens é mais adequada para simular esse tipo de sistema. A complexidade dos dois modelos foi comparada de acordo com quatro parâmetros: número de blocos, recursos computacionais, elementos de SBA e linhas de código utilizadas.

PALAVRAS CHAVE. Simulação a Eventos Discretos, Simulação Baseada em Agentes, Complexidade.

Tópicos Simulação

ABSTRACT

Simulation is a tool that allows to simplify the study of a real system by representing it through a computational environment. Among the approaches that can be used to represent a model we can point out Discrete Event Simulation (DES) and Agent-based Simulation (ABS), which can be combined to form Hybrid Simulation (HS). This study analyzes two simulation models that represent a hypothetical situation in *flow-shop* manufacture, one model being build based on DES and the other based on HS, to rate their complexity and verify which of the two approaches is the most adequate to simulate this specific system. The models were compared based in four parameters: number of blocks, computational resources, ABS elements and programming lines used.

KEYWORDS. Discrete Event Simulation, Agent-based Simulation, Complexity.

Paper topics Simulation

1. Introdução

No contexto de engenharia de produção, a área de gestão de sistemas de produção envolve a tomada de decisão nos níveis estratégico, tático e operacional [Mello et al., 2010]. Neste sentido, a simulação surge como uma ferramenta de auxílio à tomada de decisão, que pode ser utilizada para simular a operação de sistemas complexos, estudar sua performance e experimentar hipóteses [Jacob, 2013; Sharma, 2015].

A simulação permite poupar recursos econômicos, otimizar o uso do tempo, prever diversos acidentes que possam ocorrer, projetar plantas, identificar gargalos, entre outras vantagens, sendo útil em qualquer uma das fases do ciclo de vida de um sistema de manufatura. O processo de simulação consiste em imitar um sistema através do computador, colocando-o em um ambiente controlado, onde seu desempenho pode ser medido sob diversas circunstâncias sem a necessidade de atuar sobre o sistema real [Law e Kelton, 1991].

Dado que a simulação pode ser utilizada para se representar sistemas complexos, os modelos computacionais gerados por ela também tendem a se tornar eles próprios muito complexos de acordo com o sistema abordado. Segundo Chwif et al. [2000], que apontam que os modelos de simulação tendem a se tornar cada vez mais complexos com o tempo, e que isto tem trazido diversos desafios para os modeladores.

Entre as metodologias utilizadas na simulação, podemos destacar a Simulação a Eventos Discretos (SED) e a Simulação Baseada em agentes (SBA). A principal diferença entre os dois tipos de abordagem se refere a como o problema é abordado. Segundo Siebers et al. [2010], a SED possui uma abordagem *top-down*, focada na modelagem dos sistemas e como estes reagem aos eventos, sem se concentrar nas micro-estruturas que geram o processo. Já a SBA foca justamente no comportamento individual de cada elemento presente no sistema, se tratando de uma abordagem *bottom-up*. Contudo, elementos característicos dos dois tipos de abordagem podem ser combinados, gerando assim a Simulação Híbrida (SH).

Algumas pesquisas têm inserido elementos de agentes em modelos SED (gerando assim modelos de Simulação Híbrida), como Dubiel e Tsimhoni [2005], Hao e Shen [2008] e Fakhimi et al. [2014]. Acredita-se que o uso dos agentes possa permitir a redução da complexidade dos modelos, em termos do número de recursos utilizados para modelá-los, devido às possibilidades geradas pela SBA. Dentro deste contexto, este trabalho tem como objetivo avaliar a contribuição da Simulação Baseada em Agentes na redução da complexidade em modelos de simulação, através da contagem de recursos utilizados por modelo para medir dois modelos de simulação que representam o mesmo sistema (um *flow-shop* com três postos de trabalho), sendo um deles construído através da SED e o outro através da SH.

2. Revisão Bibliográfica

2.1 Simulação a Eventos Discretos

Na definição de Schribber et al. [2014], uma simulação a eventos discretos é caracterizada por modelos onde as mudanças só podem ocorrer em determinados pontos no tempo (aleatórios ou não), chamados eventos. Jacob [2013] aponta um dos fatores-chave que diferenciam a SED de outras abordagens de simulação, que são as restrições que limitam em como o sistema pode sofrer mudanças ao longo do tempo. Cada mudança ocorre instantaneamente, em intervalos bem definidos de tempo.

Os principais elementos a serem representados na SED de acordo com Schribber et al. [2014] são:

- Entidades:** são os itens processados pelo sistema, respondendo aos eventos (matéria-prima, informação, clientes, etc.);

- Recursos:** quaisquer elementos do sistema que realizam algum serviço (geralmente sobre as entidades). Quase sempre possuem limitações de capacidade, e as entidades em fluxo competem pelo seu uso;

-Controles: são elementos criados para permitir atrasos e alternativas lógicas no fluxo da simulação, como contadores, variáveis, desvios e outros;

-Operações: cada um dos passos por onde a entidade pode passar dentro do sistema.

Siebers et al. [2010] apontam que a Simulação a Eventos Discretos vem sendo há mais de 40 anos o principal pilar da comunidade científica em trabalhos que envolvem simulação na Pesquisa Operacional. No entanto, alguns autores pontuam determinados pontos fracos desta abordagem, principalmente onde há a necessidade de se representar de forma mais precisa o comportamento humano em um sistema. Elkosantini [2015] pontua que uma das causas para essa imprecisão na simulação do fator humano vem do fato dela se utilizar frequentemente de dados determinísticos para representar o comportamento humano nos modelos computacionais.

2.2. Simulação Baseada em Agentes

A Simulação Baseada em Agentes (ou SBA) é uma abordagem de simulação que tem como foco representar o comportamento dos “agentes” que são autônomos, individuais e podem interagir entre si [Macal e North, 2013]. Segundo Epstein e Axtell [1996] a SBA consiste em três elementos principais: os agentes, suas regras e o ambiente. A SBA possui uma demanda crescente visto que há a necessidade de se analisar sistemas cada vez mais complexos, principalmente em termos de interdependências entre elementos [Macal e North, 2009].

A SBA aparece como uma proposta para melhor representar o elemento humano em modelos de simulação. Para Macal [2016], a SBA ainda está longe de ser compreendida ou aceita, mas esta possui características que complementam outros tipos de simulação. Para Dubiel e Tsimhoni [2005], a SBA fornece soluções mais precisas para a modelagem de sistemas que precisam representar a tomada de decisões de seres humanos, ou padrões de movimento muito detalhados, já que seu objetivo é simular entidades autônomas e inteligentes e suas interações entre si e com o ambiente.

Para Macal e North [2009], O termo “agentes” não possui uma definição universalmente aceita, mas os autores apresentam alguns elementos que definem o termo:

-São autônomos;

-São modulares (individuais);

-São sociais, e interagem com outros agentes.

Macal [2016] propôs quatro definições para tipos de agentes, na ordem crescente de complexidade, representados no Quadro 1.

Tipo	Individualidade	Comportamentos	Interações	Adaptabilidade
Tipo I	Agentes heterogêneos individuais	Prescrito, roteirizado	Limitada	Não há
Tipo II	Agentes heterogêneos individuais	Autônomo, dinâmico	Limitada	Não há
Tipo III	Agentes heterogêneos individuais	Autônomo, dinâmico	Com outros agentes e com o ambiente	Não há
Tipo IV	Agentes heterogêneos individuais	Autônomo, dinâmico	Com outros agentes e com o ambiente	Agentes mudam de comportamento durante a simulação

Quadro 1 – Classificação do Tipos de Agentes.

Fonte: Adaptado de [Macal 2016]

2.3. Complexidade

Segundo ElMaraghy et al. [2012], o significado do termo “complexidade” é ambíguo, e não existe uma definição precisa para ele. Para os autores, seguindo uma definição mais próxima do dicionário Oxford um sistema pode ser considerado mais complexo se mais elementos e relações entre esses elementos existirem. Além disso, cada área das ciências e engenharia define e enxerga a complexidade de formas diferentes.

Entre diversas definições e conceitos relacionados à complexidade, podem ser citados: aspecto cognitivo, número de elementos e as relações entre eles, incertezas, entre outros [Brooks e Tobias, 1996, Chwif et al., 2000, Elmaraghy et al., 2012]. Para Popovics e Monostori [2016], o termo “complexidade” é tipicamente utilizado em associação com sistemas, compostos de componentes que podem ser dependentes ou independentes uns dos outros, e a complexidade cresce à medida em que se aumenta o número de elementos e as conexões entre eles.

Em simulação, busca-se representar um sistema da forma menos complexa possível, desde que isso não torne a representação computacional inválida. Ahmed et al. [2016] apontam que tamanho e complexidade são dois dos mais importantes fatores que afetam a prática da modelagem e simulação. Chwif et al. [2000] apontam que a complexidade dos modelos de simulação tem impacto não só no processamento computacional, mas em outros aspectos como a gestão de projetos de simulação, comunicação, identificação das restrições, entre outros. Os autores também propõem que se busque formas de reduzir a complexidade dos modelos ao invés de se tentar lidar com modelos muito complexos.

Ahmed et al. [2016] realizaram uma pesquisa perguntando para 20 profissionais de simulação quais elementos de um modelo servem de medição para o seu tamanho e complexidade. As respostas divergiram, porém alguns elementos foram citados por diversos pesquisadores, como número de blocos (30% do total de entrevistados), número de variáveis (30%), tempo necessário na modelagem (25%), linhas de código utilizadas (15%) e número de atividades, processos e passos (55%).

3. Metodologia

Essa pesquisa tem como objetivo contribuir com a discussão sobre a utilização da Simulação Híbrida, especificamente o uso de agentes em modelos de Simulação a Eventos Discretos, e como o seu uso pode reduzir a complexidade dos modelos de simulação. Para isso optou-se por um Estudo de Caso envolvendo dois modelos empíricos. Assim, segundo as definições de Bertrand e Fransoo [2002], esta pode ser classificada como uma pesquisa Empírica Descritiva, pois busca auxiliar na melhor compreensão do processo de modelagem e simulação, esclarecendo para os usuários da simulação em quais situações seria mais viável se optar por cada abordagem.

A primeira etapa do trabalho constituiu na elaboração de dois modelos de simulação distintos, porém representando o mesmo sistema, um processo de manufatura *flow-shop* com três postos de trabalho. Enquanto o primeiro modelo foi elaborado através da SED (elemento humano representado como um recurso e com comportamento definido a partir de eventos esperados), o segundo se utilizou da SH (elemento humano representado por agentes, cujos comportamentos foram definidos a partir de um diagrama de estado, onde a autonomia dos agentes se encaixa no Tipo 1 definido por Macal [2016]).

Para a validação dos modelos, uma análise foi feita a partir do software Minitab®, para garantir que não se poderia rejeitar a hipótese nula (H_0) de que os resultados finais (número de peças produzidas) pelos dois modelos possuem a mesma média. O *software* utilizado para a construção dos modelos de simulação foi o Anylogic®, já que este permite a construção de modelos SED e SBA, e a combinação destas duas abordagens para gerar modelos híbridos (SH).

A etapa seguinte consistiu na avaliação da complexidade dos dois modelos partindo-se da contagem de quatro tipos de elementos utilizados em ambos os modelos. Esta avaliação não levou em consideração outras etapas do processo de modelagem e simulação, considerando apenas a quantidade de recursos necessários na construção dos modelos computacionais. Os elementos escolhidos para comparar os modelos foram o número de blocos, de recursos adicionais, elementos da SBA e linhas

de código. Estes elementos foram definidos para servir de condições de contorno, possibilitando comparar a complexidade dos dois modelos, que representam duas alternativas para se representar um mesmo sistema, de acordo com a proposta de Brooks e Tobias [1996]. O tempo de construção dos modelos não foi considerado, por ser uma medida subjetiva, e ambos os modelos foram construídos pelos autores deste trabalho (a comparação poderia ser invalidada se cada modelo fosse construído por um modelador com habilidades diferentes).

4. Desenvolvimento

4.1. Concepção do Modelo

A primeira etapa necessária para o desenvolvimento da pesquisa é a concepção dos modelos a serem simulados. Ambos representam um mesmo sistema que consiste em uma linha de produção que trabalha uma peça através de três estações de trabalho, na qual há uma fila com capacidade 1 entre cada um deles. A peça deve ser inspecionada no primeiro posto, embalada no segundo e expedida no terceiro. Cada posto de trabalho possui um operador associado e estes trabalharão em turnos das 8 às 17h, com pausas para o almoço entre 12 e 13h. A função de cada funcionário é buscar a peça na fila anterior e realizar a tarefa no seu respectivo posto de trabalho. Os tempos definidos para cada etapa são distribuições normais e apresentados em segundos: o tempo de inspeção possui $\mu = 70,14$ e $\sigma = 11,18$, para a embalagem $\mu = 61,33$ e $\sigma = 8,29$, e para a expedição $\mu = 50,97$ e $\sigma = 5,18$.

4.2. Construção dos modelos computacionais

A partir da concepção apresentada, os modelos foram construídos no Anylogic®. Dois modelos foram construídos, um deles com a abordagem utilizada na Simulação a Eventos Discretos e outro na abordagem utilizada na Simulação Híbrida (elementos de SBA com elementos da SED).

Na modelagem SED, devido a abordagem *top-down*, o mais importante é fluxo da peça trabalhada e os eventos que alteram o estado do sistema, com foco no total de peças produzido. O elemento humano é tratado como um recurso associado à peça e a um posto de trabalho, e não há uma preocupação na forma como ele toma decisões individualmente. Com esta perspectiva, chegou-se no modelo da Figura 1.

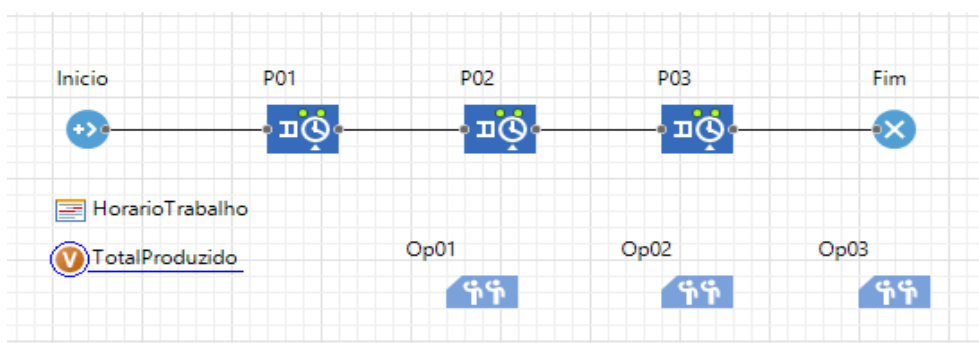


Figura 1 - Modelo SED.
Fonte: [Bernardo 2017]

Ao se usar elementos da modelagem SBA, a forma de visualizar este mesmo problema é diferente. Nesse caso a abordagem é *bottom-up*, ou seja, o comportamento de cada um dos operadores é definido individualmente, e a partir deles observa-se como o sistema se comporta. Dessa forma, obtém-se como resultado o modelo da Figura 2.

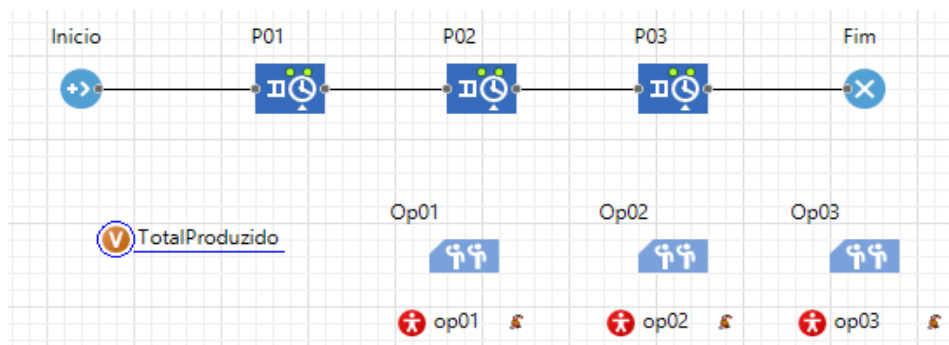


Figura 2 - Modelo SH.

Fonte: [Bernardo 2017]

Como pode-se observar, devido a partirem da mesma concepção, os modelos não sofrem grandes diferenças em sua estrutura. Neste caso, porém, o comportamento dos operadores é definido individualmente a partir de um diagrama de estado dos agentes, como mostrado na Figura 3.

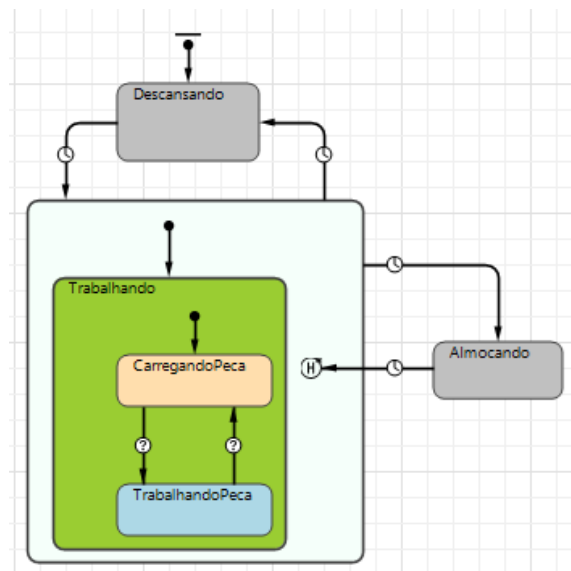


Figura 3 - Diagrama de estados dos agentes.

Fonte: [Bernardo 2017]

4.3. Comparação dos modelos

Para a comparação da complexidade apresentada em cada abordagem, primeiramente é necessário que os modelos computacionais estejam coerentes com o modelo conceitual proposto. Como o modelo não parte de uma situação existente, não há como validá-lo baseado em um padrão real. Porém, ao partir de uma mesma situação hipotética, é esperado que nas duas situações o número de peças produzidas seja próximo. Foi realizado um teste estatístico com os valores extraídos das duas simulações. Por se tratar de dados advindos de fontes onde se desconhece a média e o desvio padrão populacionais utilizou-se o teste de hipóteses do tipo *two-sample t* para comparar as médias das duas populações, com hipóteses: H_0 = as médias são iguais e H_1 = as médias são diferentes, com um nível de significância de 0,05 [Montgomery, 2003].

Foi utilizado o *software* Minitab® para se obter o número de amostras para o teste, que resultou em 50 amostras de cada modelo. Estes dados se encontram nos Anexos A e B. O teste resultou em um *p-value* de 0,949, o que significa que não se pode rejeitar a hipótese de que essas médias são

iguais, ou seja, pode-se considerar que os modelos estão atingindo resultados estatisticamente bem próximos.

4.4. Análise da complexidade

Com os modelos completos e chegando nos mesmos resultados, foi possível afirmar que eles representam a mesma situação. A partir disso, foram analisados os recursos computacionais utilizados para construir cada modelo.

Os recursos foram divididos em:

- **Blocos:** Recursos pré-estruturados no Anylogic®, a partir dos quais se pode construir o modelo. Estes podem ser derivados da SED, da SBA ou mesmo da SD (dinâmica de sistemas). Blocos podem representar operações, recursos, filas, entre outros. A Figura 4 apresenta os principais blocos utilizados no software Anylogic® para a construção de modelos de simulação.
- **Recursos Adicionais:** Funções adicionadas para auxiliar o controle das fluxos e decisões dentro do modelo, como variáveis, cronogramas, conexões e diagrama de agentes.
- **Agentes:** Recursos específicos da abordagem SBA, como os diagramas de estado dos agentes.
- **Linhas de programação:** Número de linhas de programação necessárias para o funcionamento de cada modelo. A forma de medir programas orientados a objetos através do número de linhas de código está de acordo com Borys & Milosz (2010).

Ilustração	Nome	Função
	Source (Entrada)	Inicia o modelo e gera entidades
	Queue (Fila)	Armazena entidades que não podem ser processadas no processo posterior
	Service (Serviço)	Captura, protela e libera entidades
	Delay (Protelação)	Protela entidades
	Sink (Saída)	Finaliza o modelo e absorve as entidades processadas
	Resource Pool (Conjunto de recursos)	Define as unidades de recurso que possuem a capacidade de capturar e de liberar entidades dentro dos elementos Seize, Release, Assembler e Service
	Schedule (Cronograma)	Define um cronograma a ser seguido dentro dos elementos Resource pool, Source e PedSource
	Variables (Variáveis)	Armazena resultados do modelo computacional e modela, no decorrer do tempo, unidades de dados ou características do objeto
	Agent (Agente)	Tem a capacidade de representar diversos componentes presentes em sistemas reais através de atributos e comportamentos
	Parameter (Parâmetro)	Descreve o objeto modelado representando as características do mesmo
	Event (Evento)	Usado para alterar algo no modelo em determinado período de tempo ou sob alguma condição

Figura 4 - Blocos do Anylogic®.
Fonte: Adaptado de [Barbieri 2016].

Os dois modelos utilizaram quantidades diferentes de recursos computacionais, por terem sido construídos a partir de abordagens diferentes. No primeiro modelo, construído a partir da Simulação a Eventos Discretos pura, foram utilizados blocos para a entrada e saída do modelo, três postos de trabalho (P01, P02 e P03) e três operadores (Op01, Op02 e Op03), vide Figura 1. Além disso foram usadas três funções como recursos adicionais, no caso a variável *OnOff* (serve para auxiliar na apresentação visual, alterando a animação dos operadores caso estes estejam trabalhando ou parados), a variável *TotalProduzido*, que armazena o valor total de peças produzidas ao fim da simulação e o cronograma *Horário Trabalho* que está associado aos recursos e define quais horários os operadores devem trabalhar.

No caso do modelo feito a partir da Simulação Híbrida, foram usados os mesmos blocos de anteriormente e mais três, referentes aos agentes (op01, op02 e op03), vide Figura 2. Também se adicionou os três agentes (Op01, Op02 e Op03), e o diagrama de agentes apresentado na Figura 3. Foram mantidas a variável *TotalProduzido* e o cronograma *Horário Trabalho*, e foram acrescentadas a variável *IsConnected* (auxilia na lógica que controla as ações dos agentes) e o cronograma *Horário Almoço* (auxilia na programação, separando os turnos de trabalho entre antes e depois do almoço).

Além destes recursos mencionados, também foram contados o número de nós e conectores em cada modelo, além do uso de linhas de programação. O número de conexões e nós se manteve constante nos dois casos (4 conexões e 13 nós). Com relação aos recursos de programação, observou-se que o primeiro modelo (SED) utilizou 7 recursos de programação, a maioria com o objetivo de se auxiliar na animação do modelo em 3D. No caso do modelo com os agentes temos também o diagrama de agentes (Figura 3) onde a programação é executada.

Os recursos de programação utilizados no segundo modelo foram divididos em três conjuntos: o primeiro possui 6 linhas de código idênticas às do primeiro modelo, com o objetivo de auxiliar a animação e calcular o total produzido; O segundo possui 6 linhas de código que representam a lógica das ações do agente e o terceiro possui 12 linhas de código que controlam as transições que ocorrem dentro do diagrama de estados do agente (Figura 3).

4.5 Comparação da Complexidade

Para se comparar a complexidade dos dois modelos, a análise foi feita com base no número de adições e alterações que foram necessárias para se chegar no resultado. As Tabelas 1 e 2 apresentam os valores numéricos associados às adições e alterações para cada uma das 4 categorias: **blocos, recursos adicionais, agentes e linhas de programação** para cada um dos modelos.

SIMULAÇÃO A EVENTOS DISCRETOS				
	Blocos	Recursos Adicionais	Agentes	Linhas de Programação
Início	6	1	0	0
P01	9	3	0	2
P02	9	3	0	1
P03	9	3	0	2
OP01	6	2	0	2
OP02	6	2	0	2
OP03	6	2	0	2
FIM	1	0	0	1
TOTAL	52	16	0	12

Tabela 1 - Uso de recursos para o primeiro modelo.

Fonte: [Bernardo 2017]

SIMULAÇÃO HÍBRIDA				
	Blocos	Recursos Adicionais	Agentes	Linhas de Programação
Início	6	1	0	0
P01	9	2	0	20
P02	9	2	0	19
P03	9	2	0	20
OP01	3	1	4	0
OP02	3	1	4	0
OP03	3	1	4	0
FIM	1	0	0	1
TOTAL	43	10	12	60

Tabela 1 - Uso de recursos para o primeiro modelo.

Fonte: [Bernardo 2017]

Pelas tabelas é possível perceber que a SED necessitou de mais parâmetros e recursos adicionais para programar os três operadores (OP01, OP02 e OP03). Na abordagem híbrida a situação é diferente: os parâmetros e recursos adicionais dos operadores foram substituídos, já que estes pararam de ser tratados como recursos e passaram a ser tratados como agentes. Apesar da capacidade de tomar decisões desses agentes ser bem limitada (pois se trata de agentes tipo 1), estes possuem seu próprio diagrama de estados e seu conjunto de regras. Para estabelecer este conjunto de regras foi necessário um número maior de linhas de programação e conhecimento de Programação Orientada a Objetos.

5. Conclusões

O presente estudo buscou fazer uma análise da complexidade relacionada à diferentes abordagens dentro da simulação tendo como base um estudo de uma situação hipotética dentro do ramo da manufatura. Neste estudo buscou-se avaliar qual seria o método mais adequado de simulação entre SED e SH para o caso onde os recursos do sistema possuem comportamento individual e roteirizado, sem reagir a situações novas ou alterar seu comportamento.

É importante ressaltar que na análise de situações reais a complexidade de um modelo pode ser agravada por diversos fatores que vão além da construção do modelo computacional. Dentre eles, pode-se citar a dificuldade de entender o sistema real, a inabilidade de se modelar corretamente o problema, a inabilidade de transferir o modelo conceitual para um modelo computacional equivalente, falta de conhecimento dos softwares e objetivos mal definidos [Chwif et al., 2000].

No presente estudo, porém, a situação era hipotética. Isso fez com que esses fatores não fossem levados em conta e a pesquisa tratou apenas da comparação relacionada ao número de recursos computacionais utilizados em cada abordagem. A presente comparação foi possível devido as condições de contorno estabelecidas, já que foi isolada a parte computacional do desenvolvimento da simulação, ignorando a complexidade envolvida em outras etapas como a coleta de dados e a modelagem conceitual.

A partir desta comparação foi possível verificar que, para este tipo de situação, onde o agente é reativo (tipo 1 de Macal [2016]), há uma menor complexidade associada quando a SED é utilizada de forma isolada. Por esse agente possuir um comportamento mais simples, ele pode ser modelado a partir de funções já pré-programados dentro do software de simulação, exigindo-se um menor número de recursos computacionais e menor conhecimento específico em programação.

Mesmo se tratando de um agente com comportamento roteirizado, a sua construção na SH, com diagramas de estado, exige que os responsáveis pela simulação possuam um conhecimento em linguagem de programação orientada a objetos, neste caso linguagem JAVA. Além disso, na programação normalmente há um maior número de formas e estruturas de código que podem chegar num mesmo resultado, e isso faz com que um simulador com menor conhecimento de programação possa criar estruturas demasiadamente complicadas para representar uma tomada de decisão, aumentando ainda mais a complexidade.

Ainda que esta conclusão seja válida para a situação apresentada, não é possível afirmar que essa característica se mantenha em agentes com maior autonomia e necessidade de tomar decisões mais elaboradas. Em tais situações imagina-se que o número de recursos adicionais e parâmetros necessários para representar o operador aumente proporcionalmente à complexidade deste, e nestes casos o uso da SH ou da SBA pode ser necessário para simplificar e reduzir a complexidade do modelo.

A compreensão dos fatores relacionados à complexidade pode otimizar a tomada de decisão por parte de gestores e pesquisadores no ramo da simulação, auxiliando-os a optar pela abordagem de simulação mais adequada. Ainda assim, o tema é pouco explorado na comunidade científica e há poucos conceitos relacionados à definição e mensuração de complexidade.

Futuros estudos serão desenvolvidos visando aprimorar esta metodologia de medição de complexidade apresentada neste artigo e ela será utilizada para comparar outros pares de modelos, um para cada tipo de agente proposto por Macal [2016], onde espera-se poder mensurar a contribuição da abordagem SBA na redução da complexidade de modelos de simulação que representem agentes com outras capacidades, como a de interagirem entre si, mudarem de estado durante a simulação e se adaptarem.

Agradecimentos

Os autores desse artigo gostariam de expressar gratidão à UNIFEI, CAPES, CNPq e FAPEMIG pelo apoio concedido ao desenvolvimento dessa pesquisa.

Referências

- Ahmed, R., Shah, M. e Uhmar, M. (2016). Concepts of Simulation Model Size and Complexity. *International Journal of Simulation Modelling*, 15(2):213-222.
- Barbieri, J. P. (2016). Análise da Representação do Fator Humano Presente em um Sistema de Manufatura Através da Simulação Híbrida. 102 p. Dissertação (Mestrado Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, Itajubá, MG, UNIFEI.
- Bernardo, M. M. (2017). Comparação da Complexidade de Modelos Híbridos e Discretos no Ramo da Manufatura. 33 f. Dissertação (Mestrado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, Itajubá, MG, UNIFEI.
- Bertrand, J. W. e Fransoo, J. C. (2002). Modelling and Simulation: operations management research methodologies using quantitative modeling. *International Journal of Operations & Production Management*, 22(2):241-264.
- Bonabeau, E. (2002). Agent-based modeling: methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences*, 99(3):7280-7287.
- Borys, M.; Milosz, M. (2010). Advanced Object-Oriented Technology. PIPIS.
- Brighenti, J. R. N. (2006). Simulação e otimização de uma linha de manufatura em fase de projeto. 116 f. Dissertação (Mestrado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção, Itajubá, MG, UNIFEI.
- Brooks, R. J. e Tobias, A. M. (1996). Choosing the Best Model: Level of Detail, Complexity, and Model Performance. *Mathematical and Computer Modelling*. 24(4):1-14.
- Chwif, L., Barreto, M. R. P. e Paul, R. J. (2000). On Simulation Model Complexity. In *Winter Simulation Conference*, Orlando, FL, USA.

Dubiel, B. e Tsimhoni, O. (2005). Integrating Agent Based Modeling into a Discrete Event Simulation. In: *Winter Simulation Conference*. Orlando, FL, USA.

Elkosantini, S. (2015). Toward a new generic behavior model for human centered system simulation. *Simulation Modelling Practice and Theory*, 52:108-122.

ElMaraghy W., ElMaraghy H., Tomiyama T. e Monostori L. (2012). Complexity in engineering design and manufacturing. In: *CIRP Manufacturing Technology*.

Epstein, J. e Axtell, R. (1996). Growing Artificial Societies. The Brookings Institute, Washington, DC, USA.

Jacob, M. (2013). Discrete Event Simulation. *Resonance*, p.78-86.

Law, A. M. e Kelton, W. D. (1991). Simulation Modelling and Analysis. McGraw-Hill, New York, USA.

Macal, C. M. e North, M. J. (2013). Introductory Tutorial: Agent-Based Modeling and Simulation. In: *Winter Simulation Conference*, Washington, D. C., USA.

Macal, C. M. e North, M. J. (2009). Agent-Based Modeling and Simulation. In: *Winter Simulation Conference*. Austin, TX, USA.

Macal, C. M. e North, M. J. (2013). Introductory Tutorial: Agent-Based Modeling and Simulation. In: *Winter Simulation Conference*, Washington, D. C., USA.

Macal, C. M. (2016). Everything you need to know about agent-based modelling and simulation. *Journal of Simulation*, 10(2):144-156.

Mello, C. H. P., Miguel, P. A. C., Fleury, A., Nakano, D. N., Turrioni, J. B., Lee Ho, L., Morabito, R., Martins, R. A. e Pureza, V. (2010). Metodologia de pesquisa em engenharia de produção e gestão de operações. Campus, Rio de Janeiro.

Montgomery, D. C. e Runger, G. C. (2003). Estatística aplicada e probabilidade para engenheiros. LTC, Rio de Janeiro.

Sharma, P. Discrete-Event Simulation. (2015). *International Journal of Scientific and Technology Research*, v. 4, n. 2, p. 136-140.

Siebers, P. O., Macal, C. M., Garnett, J., Buxton, D. e Pidd, M. (2010). Discrete-event simulation is dead, long live agent-based simulation! *Journal of Simulation*, 4(3): 204-210.

Schribber, T. J.; Brunner, D. T. e Smith, J. F. (2014). Inside discrete-event simulation software:how it works and why it matters. In: *Winter Simulation Conference*, Huntington Beach, CA, USA.

Anexo A - Total de peças produzido na simulação para o modelo SED

SIMULAÇÃO A EVENTOS DISCRETOS									
1812	1819	1806	1819	1821	1800	1818	1806	1826	1819
1805	1820	1817	1813	1813	1813	1812	1818	1819	1814
1813	1807	1821	1817	1814	1814	1811	1799	1805	1817
1809	1810	1809	1815	1819	1809	1805	1811	1823	1805
1816	1808	1812	1809	1808	1814	1811	1809	1799	1804

Anexo B - Total de peças produzido na simulação para o modelo SH

SIMULAÇÃO HÍBRIDA									
1809	1793	1816	1798	1814	1801	1822	1809	1810	1806
1800	1808	1830	1817	1803	1825	1806	1819	1800	1808
1816	1807	1824	1811	1822	1820	1816	1810	1804	1806
1811	1819	1811	1830	1804	1814	1821	1795	1809	1825
1809	1811	1816	1821	1814	1796	1801	1827	1821	1813