

INTEGRANDO INTELIGÊNCIA ARTIFICIAL NO SOFTWARE ANYLOGIC® UTILIZANDO PATHMIND®

Anderson Lino de Paula Martins

Universidade Federal de Itajubá (UNIFEI)
Avenida BPS, 1303, Itajubá – MG
linoanderson93@gmail.com

Alyson Benoni Matias Pereira

Universidade Federal de Itajubá (UNIFEI)
Avenida BPS, 1303, Itajubá – MG
alysonmp@gmail.com

Yago Toledo Lima

Universidade Federal de Itajubá (UNIFEI)
Avenida BPS, 1303, Itajubá – MG
yagotoledolima@gmail.com

Alexandre Ferreira de Pinho

Universidade Federal de Itajubá (UNIFEI)
Avenida BPS, 1303, Itajubá – MG
pinho@unifei.edu.br

José Arnaldo Barra Montevechi

Universidade Federal de Itajubá (UNIFEI)
Avenida BPS, 1303, Itajubá – MG
montevechi@unifei.edu.br

Resumo

A Simulação híbrida, que no caso desse trabalho une a Simulação Baseada em Agentes com a Simulação à Eventos Discreto, vem sendo muito difundida nas últimas duas décadas, pois permite resolver problemas que outros tipos de simulação separadamente não são capazes. Com o avanço da Inteligência Artificial (IA) e a inclusão dela em modelos de simulação, vários problemas que não possuíam solução fácil começaram a ser solucionados. Esse trabalho tem como objetivo utilizar o pacote de IA PathMind Helper® em conjunto com o software de simulação AnyLogic® para desenvolver um modelo de um problema que possa ser solucionado com o auxílio da IA. A ferramenta aparenta ser promissora e trouxe resultados satisfatórios. Quando a simulação foi realizada sem a utilização da IA treinada, a taxa de acerto da ferramenta foi de aproximadamente 12,5%, enquanto após o treinamento a simulação passou a apresentar uma taxa de acerto 100% para esse problema.

Palavras-chaves: Simulação Baseada em Agentes, Simulação Híbrida, AnyLogic, Inteligência Artificial, Pathmind Helper.

Tópicos: Simulação, inteligência Artificial.

Abstract

Hybrid Simulation, that in this paper unites Agent-based Simulation with Discrete Event Simulation, has been widespread in the last two decades, because it allows solving problems that both simulation models cannot do separately. With the advance of Artificial Intelligence (AI) and its inclusion in simulation models, several problems that did not have solution began to be solved. This paper has as objective the use of PathMind Helper® Artificial Intelligence package together with simulation software AnyLogic® in order to develop a model of a problem that can be solved with the help of AI. The PathMind Helper® seems to be very promising and presents an extremely satisfying result. When the simulation were performed without using the trained Artificial Intelligence, the accuracy rate of the tool were approximately 12.5%, while after the training the simulation presented an accuracy rate of 100% for this problem.

Keywords: Agent-based Simulation, Hybrid Simulation, AnyLogic, Artificial Intelligence, PathMind Helper.

Paper topics: Simulation, Artificial Intelligence.

1. Introdução

O termo indústria 4.0 surgiu na Feira de Hannover, na Alemanha, em 2011 e anunciada oficialmente como estratégia do país para um pioneirismo industrial, que atualmente está revolucionando o setor de manufatura, uma vez que estamos no auge da quarta revolução industrial [Xu et al. 2018]. Os autores [Liao et al. 2017] corroboram com essa informação, pois afirmam que essa revolução tem atraído cada vez mais atenção ao redor do mundo nos últimos anos. E [Ferreira et al. 2020] concluem que há uma tendência de aumento nas pesquisas relacionadas a Indústria 4.0 e a simulação, e que a simulação híbrida está entre as principais abordagens nesse contexto.

A simulação híbrida é a união de dois ou mais tipos de simulação [Swinerd e McNaught 2012] e no contexto desse trabalho envolve a utilização da Simulação Baseada em Agentes, pois cada caixa é considerada um agente que possui características próprias, e a Simulação à Eventos Discretos, que descreve o processo como um todo desde o modelo conceitual, utilizando o IDEF-SIM, até o modelo computacional que mostra como a separação ocorre.

A Inteligência Artificial (IA) e a simulação são dois ramos da pesquisa acadêmica que se desenvolveram separadamente, mas que se inter-relacionam em diversos aspectos [Mavrovouniotis 2012]. Seja quando a simulação é utilizada para desenvolvimento de ferramentas de inteligência artificial, como explica Kober *et al.* [2013], ou quando a inteligência artificial entra como elemento da simulação [Mavrovouniotis 2012]. Existem também os conceitos introduzidos pela Indústria 4.0, como os Gêmeos Digitais, que visam, entre outras coisas, unir a simulação a modelos de inteligência artificial desde sua concepção [Poddar 2018].

Entretanto, existem algumas dificuldades nessa integração, segundo [Wang et. al. 2018] conforme a estrutura da IA se torna mais complexa, o treinamento do modelo toma mais tempo e os parâmetros de otimização se tornam mais confusos, levando inclusive a problema de otimização. Um segundo obstáculo é descrito por [Ferreira et. al. 2020] e diz respeito a coleta de dados para os sistemas de simulação e IA. Os autores dizem que coletar dados do sistema real que sejam úteis para a simulação é frequentemente trabalhoso e consome muito tempo.

O objetivo deste artigo é utilizar uma ferramenta de IA juntamente com a Simulação Baseada em Agentes e demonstrar como estas duas técnicas podem ser utilizadas em conjunto. Pretende-se que este artigo seja uma leitura inicial para todos os pesquisadores e analistas de simulação que queiram combinar IA e Simulação em seus trabalhos. Neste artigo, a IA, integrada ao modelo de simulação, deverá identificar as cores das caixas criadas internamente durante a execução da simulação.

O restante do trabalho se organiza da seguinte forma: a seção 2 traz um referencial teórico sobre simulação e inteligência artificial. A seção 3 define o método de pesquisa para esse trabalho. A seção 4 traz a definição do problema, juntamente com a descrição dos modelos conceituais e computacionais bem como uma breve explicação sobre o *PathMind Helper*®. A seção 5 mostra os testes e os resultados dos mesmos. Por fim, a seção 6 conclui o trabalho e descreve alguns possíveis trabalhos futuros.

2. Referencial Teórico

2.1. Simulação Computacional

[Montevecchi *et al.* 2007] afirmam que a simulação computacional é a importação do mundo real para o mundo virtual, onde o comportamento do ambiente pode ser estudado em diversas situações, podendo ser alterado sem envolver custos e/ou riscos. A simulação é uma técnica capaz de, a partir de um modelo computacional, reproduzir um sistema real, permitindo que esse seja testado em diferentes condições [Baines, *et al.* 2004]. Além disso, [Ingalls 2008] também descreve a simulação como sendo o processo de *design* de um modelo dinâmico de um sistema atual, sendo assim possível compreender o comportamento do sistema, avaliando suas variações estratégicas.

Devido à dificuldade de conseguir definir exatamente o que é a simulação, os autores [Chwif e Medina 2015] se propuseram a dizer o que não é a simulação. Segundo os autores, a simulação não é uma bola de cristal capaz de prever o futuro, também segundo os autores ele não é um modelo matemático, apesar de utilizar fórmulas matemáticas. Além disso, os autores escrevem que a simulação não é uma ferramenta estritamente de otimização, mas sim uma ferramenta capaz de analisar diferentes cenários ao invés não traz uma resposta ótima e também não é capaz de substituir o pensamento humano na tomada de decisões. Vale lembrar que os autores também descrevem que a simulação não é uma ferramenta capaz de resolver todos os problemas.

[Standridge 2000] defende a importância da simulação computacional, pois é uma técnica comumente utilizada por engenheiros e gestores para lidar com os problemas do cotidiano operacional. O mesmo autor ainda afirma que um sistema de simulação fornece informações sobre alterações dinâmicas de um sistema.

Existem diferentes tipos de simulação, sendo três principais: sistemas dinâmicos (SD), simulação a eventos discretos (SED) e simulação baseada em agentes (SBA) [Borshchev e Filippov 2004]. Os mesmos autores ainda completam afirmando que a SD e a SED são mais tradicionais, enquanto a SBA é relativamente nova.

2.2. Simulação a Eventos Discretos

A SED é bastante utilizada desde os anos 90 na melhoria de processos de negócio, ela envolve a reprodução de problemas reais em modelos computacionais, a fim de explorar a alteração de cenários possíveis [Melão e Pidd 2006]. [Borshchev e Filippov 2004] definem a SED como uma técnica de modelagem baseada no conceito de entidades, recursos e gráficos que descrevem um fluxo de entidades e compartilhamento de recursos. Segundo [Schriber e Brunner 2011], a base para a SED é provida a partir da visão geral do fluxo, na qual consiste de unidades discretas que movem o fluxo de um ponto até o outro do sistema, no qual o estado do modelo se altera apenas em um ponto de tempo discreto da simulação, que possivelmente é aleatório.

A SED é composta pelas entidades, recursos, elementos de controle e as operações. A entidade é utilizada para designar uma unidade do tráfego e é instigada a responder aos eventos. O recurso é designado a um elemento do sistema que fornece serviço, o elemento de controle designa uma construção que suporta outros tipos de atraso ou alternativa lógica com base no estado do sistema. Por fim, a operação é uma etapa realizada por uma entidade enquanto ela se move através do sistema [Schriber e Brunner 2011].

Segundo [Siebers *et al.* 2010], os modelos de SED possuem certos atributos como: ser orientado ao processo, possuir uma técnica de modelagem de cima para baixo, ser centralizado (fio único de controle), possuir entidades passivas, ter filas como elemento chave, possuir um fluxo de entidades através de um sistema e os dados de entrada serem geralmente baseados em uma coleta/medição de dados (objetivos).

2.3. Simulação Baseada em Agentes

A SBA é uma técnica de modelagem relativamente nova, que ganhou maior atenção a partir da década de 90. Isso é comprovado devido ao aumento no número de trabalhos publicados, nos quais busca-se evidenciar o comportamento humano e social [Macal e North 2005].

Ainda, segundo [Macal e North 2005], não há uma resposta universal sobre o que é o agente em termos da SBA, mas os autores consideram que os agentes podem possuir certas propriedades e atributos, como o fato do agente ser autônomo e auto-direcionado, os agentes são modulares ou independentes e é sociável, interagindo com outros agentes. Eles também podem possuir propriedades adicionais, como o fato de que devem viver em um meio ambiente, possuir objetivos específicos que conduzem seus comportamentos, terem a habilidade de aprender e se adaptar no ambiente de acordo com suas experiências e comumente possuir atributos de recursos que indicam seu nível atual de estoque e demais recursos.

[Siebers *et al.* 2010] descrevem as situações em que é sugerido a utilização da SBA, entre elas quando o problema tem uma representação natural como agentes e não é possível representar o problema de outra forma e quando os agentes se relacionam entre si. Além disso, os autores também sugerem a utilização da SBA quando for importante que o agente tenha aspectos espaciais ou geo-espaciais nos comportamentos e quando os mesmos tenham a necessidade de aprender ou se adaptar. Os autores também descrevem que é interessante a utilização da SBA quando for importante modelar a cooperação ou organização entre agentes.

[Siebers *et al.* 2010] ainda completam destacando os atributos que definem um modelo de SBA, que segundo eles são: baseado em indivíduos, uma técnica de modelagem de baixo para cima, é descentralizado (cada agente possui sua própria linha de controle), possui entidades ativas, não possui o conceito de filas, não possui o conceito de fluxo e as distribuições de entrada são geralmente baseadas em teorias ou dados subjetivos.

2.4. Simulação Híbrida

Segundo [Brailsford 2019], a popularidade da simulação híbrida tem aumentado exponencialmente nas últimas duas décadas. A autora define a simulação híbrida como sendo a combinação de dois ou mais tipos de simulação, entre eles se encontram a Simulação Baseada em Agentes (SBA) e a Simulação a Eventos Discretos (SED). [Swinerd e McNaught 2012] defendem que uma explicação para o uso da simulação híbrida é que os analistas de pesquisa operacional confrontam cada vez mais com problemas complexos que seriam desafios para as técnicas tradicionais de simulação, com metodologia única.

[Dubiel e Tsimhoni 2005] utilizaram da simulação híbrida na implementação de um modelo humano através de um sistema de SED. Os autores consideraram a implementação de uma técnica SBA na descrição do comportamento humano como necessária, uma vez que apenas com a técnica da SED não seria possível representar tão fielmente.

2.5. Inteligência Artificial

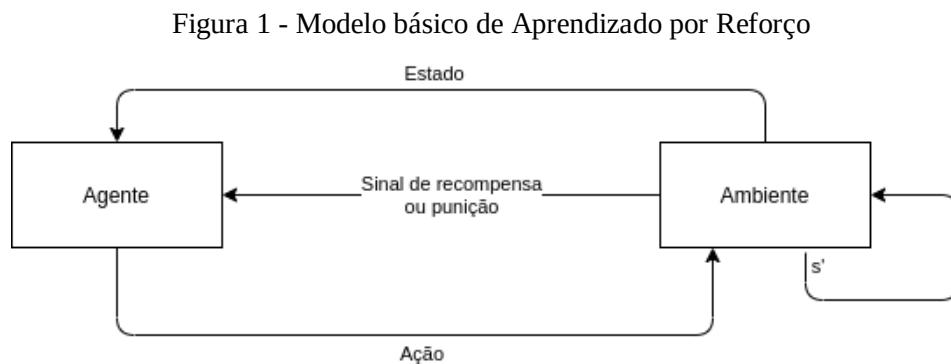
[Nilsson 2014] define a inteligência artificial como um sistema capaz de realizar algumas das atividades mentais que os seres humanos são capazes, como por exemplo resolver problemas matemáticos, compreender linguagens e pilotar veículos. Embora a inteligência artificial seja

comumente utilizada para reproduzir atividades humanas, ela não se restringe a métodos que são observados no mundo biológico [Mccarthy 1998].

A habilidade de aprender é uma das mais fundamentais do comportamento inteligente, à reprodução dessa habilidade dentro da inteligência artificial dá-se o nome Aprendizagem de Máquina [Mitchell 2013]. Uma das muitas formas de aprendizado é através da interação com o ambiente e subsequente coleta de *feedback* a partir dela. Seres humanos estão constantemente observando a correlação entre suas ações e os resultados produzidos para obter conhecimento. No mundo artificial, essa observação pode ser quantificada na forma de uma recompensa numérica que incentiva ou desestimula determinado comportamento do agente. Essa técnica é chamada de Aprendizagem por Reforço, do inglês *Reinforcement Learning* [Sutton e Barto 2018], e é explicada em mais detalhes na seção 2.5.1.

2.5.1. Aprendizado por Reforço (*Reinforcement Learning*)

A aprendizagem por reforço ocorre quando um agente precisa aprender através de tentativa e erro em um ambiente dinâmico [Kaelbling *et al.* 1996]. A Figura 1 apresenta um modelo básico para aprendizado por reforço:



Fonte: Adaptado de [Qiang & Zhongli 2011]

O modelo de Aprendizado por Reforço representado na Figura 1 se resume à repetição dos quatro passos abaixo [Qiang e Zhongli 2011]:

1. O agente primeiro observa um estado;
2. A partir deste estado realiza uma ação;
3. Essa ação produz uma mudança no estado do ambiente (s');
4. O agente observa essa atualização no estado do ambiente e calcula o valor da recompensa ou punição sobre a ação que tomou.

O objetivo de um algoritmo de Aprendizagem por Reforço é encontrar uma estratégia que produza a maior recompensa cumulativa possível. Alguns dos principais algoritmos desenvolvidos com esse propósito são: o algoritmo de Sarsa, o algoritmo da diferença temporal, o algoritmo de aprendizagem Q e o algoritmo de aproximação de função [Qiang e Zhongli 2011].

A aprendizagem por reforço promete ser uma forma de programar agentes sem que seja necessário especificar como a tarefa vai ser realizada [Kaelbling *et al.* 1996]. Nesse contexto, o treinamento de um agente em um ambiente simulado a partir de um modelo construído por especialistas, pode ser uma forma de embutir nesse agente o conhecimento dos mesmos [Deist *et al.* 2019].

3. Método de pesquisa

O trabalho utilizará o método de pesquisa de modelagem e simulação e seguirá as etapas propostas por [Montevecchi *et al.* 2007]. Os autores propõem que a primeira etapa da modelagem e simulação

é a de concepção, onde é proposto o modelo conceitual, em sequência é a implementação do modelo computacional e por fim, a etapa de análise e conclusões dos resultados.

Na etapa inicial será criado um modelo conceitual utilizando a técnica IDEF-SIM proposta pelos autores [Montevecchi et al. 2010], onde será mostrado a sequência que as caixas (entidades do sistema) percorrerão pelo sistema, desde sua concepção, até a saída do modelo. Depois será criado o modelo computacional desse sistema utilizando o software *AnyLogic*®. Na etapa final, será implementado a ferramenta de IA, o *Pathmind*®, para assim, ser analisado os resultados da efetividade dessa ferramenta a partir do índice de acertos de antes do treinamento do sistema comparado com o índice após o treinamento.

4. Desenvolvimento do modelo de Simulação

Nas seções a seguir são detalhados os procedimentos e o modelo de simulação produzidos.

4.1. Definição do problema

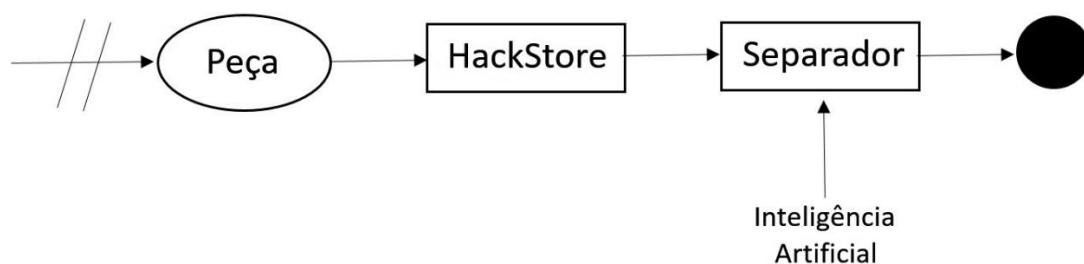
Nesse trabalho é utilizado o pacote *PathMind Helper*®, que auxilia na implementação da IA. A aplicação escolhida para o teste dessas ferramentas é uma fábrica que recebe 3 tipos diferentes de peças (as peças vermelhas, as verdes e as azuis) e a Inteligência Artificial deve ser capaz de reconhecê-las e identificar qual a peça recebida, de forma que seja possível separá-las automaticamente.

Apesar do modelo não se tratar de um problema real, ele serve como uma leitura inicial para os pesquisadores que desejam utilizar a IA em combinação com a Simulação Híbrida, pois existem problemas reais de logística, separação de materiais, entre outros, que podem ser solucionados utilizando como base esse trabalho. É importante ressaltar que por mais que os softwares de simulação sejam capazes de identificar a entidade presente no sistema, a aplicabilidade dessa pesquisa não é utilizar a simulação para identificar o objeto no sistema, mas sim, mostrar uma ferramenta que, além de solucionar o problema desse trabalho, pode ser aplicada em outras situações nas quais o sistema por si só não é capaz de resolver.

4.2. Modelo Conceitual

O modelo conceitual foi criado utilizando-se a ferramenta IDEF-SIM e a Figura 2 apresenta esse modelo. Uma peça nova chega, é inserida em um *Hack* e então separada.

Figura 2 - Modelo IDEF-SIM do projeto



Fonte: Autores

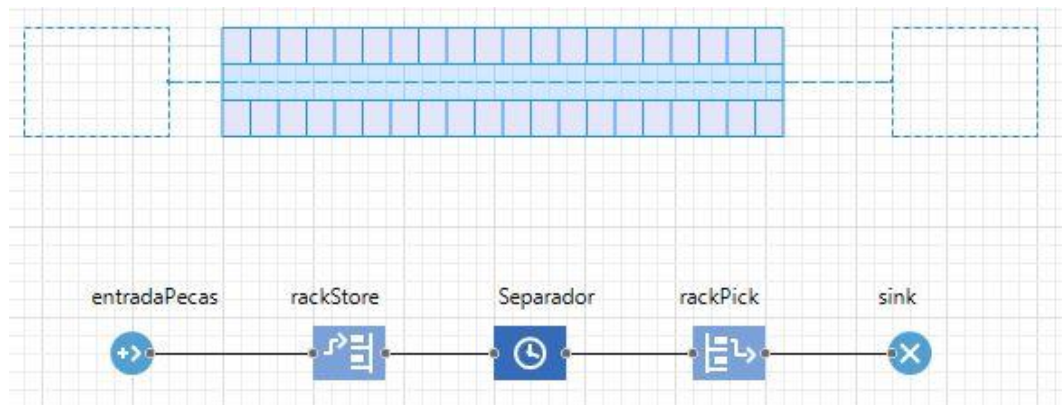
4.3. Modelo Computacional

O software *AnyLogic*® é utilizado para a criação de SBA e também para Simulação Híbrida (SED em conjunto com a SBA), sendo esse segundo o foco desse trabalho. O agente utilizado é a peça entregue para a empresa que pode ser de três tipos diferentes e o processo é o separador, que identifica o tipo de cada uma das peças recebidas.

O separador funciona utilizando-se Inteligência Artificial (IA) inserida com o auxílio do pacote *PathMind Helper*® que permite, de forma fácil, a união entre as ferramentas. A próxima seção demonstra o funcionamento do *PathMind Helper*®, juntamente com o exemplo utilizado nesse trabalho.

A Figura 3 mostra o modelo computacional criado baseado no IDEF-SIM apresentado na Figura 2. Vale ressaltar que, para melhorar a resposta da IA, é preferível limitar o tempo de cada rodada da simulação, portanto serão utilizados três cenários diferentes, 12 horas, 24 horas e 36 horas.

Figura 3 - Modelo computacional



Fonte: Autores

4.4 *PathMind Helper*®

Segundo a própria página do software [AnyLogic® 2020], o *PathMind*® permite inserir IA nos modelos de simulação e permite também monitorar e validar as variáveis que devem ser rastreadas, definir as escolhas que os agentes farão e as observações recebidas a partir do ambiente. Para utilizar esse pacote de forma gratuita é necessário possuir a versão do AnyLogic® 8.7.4., pois nas versões anteriores desse software, a ferramenta *PathMind* somente poderia ser utilizada no AnyLogic Professional.

O *PathMind Helper*® utiliza a aprendizagem por reforço (*Reinforcement Learning*) e funciona definindo-se três funções principais: observações, métricas e ações.

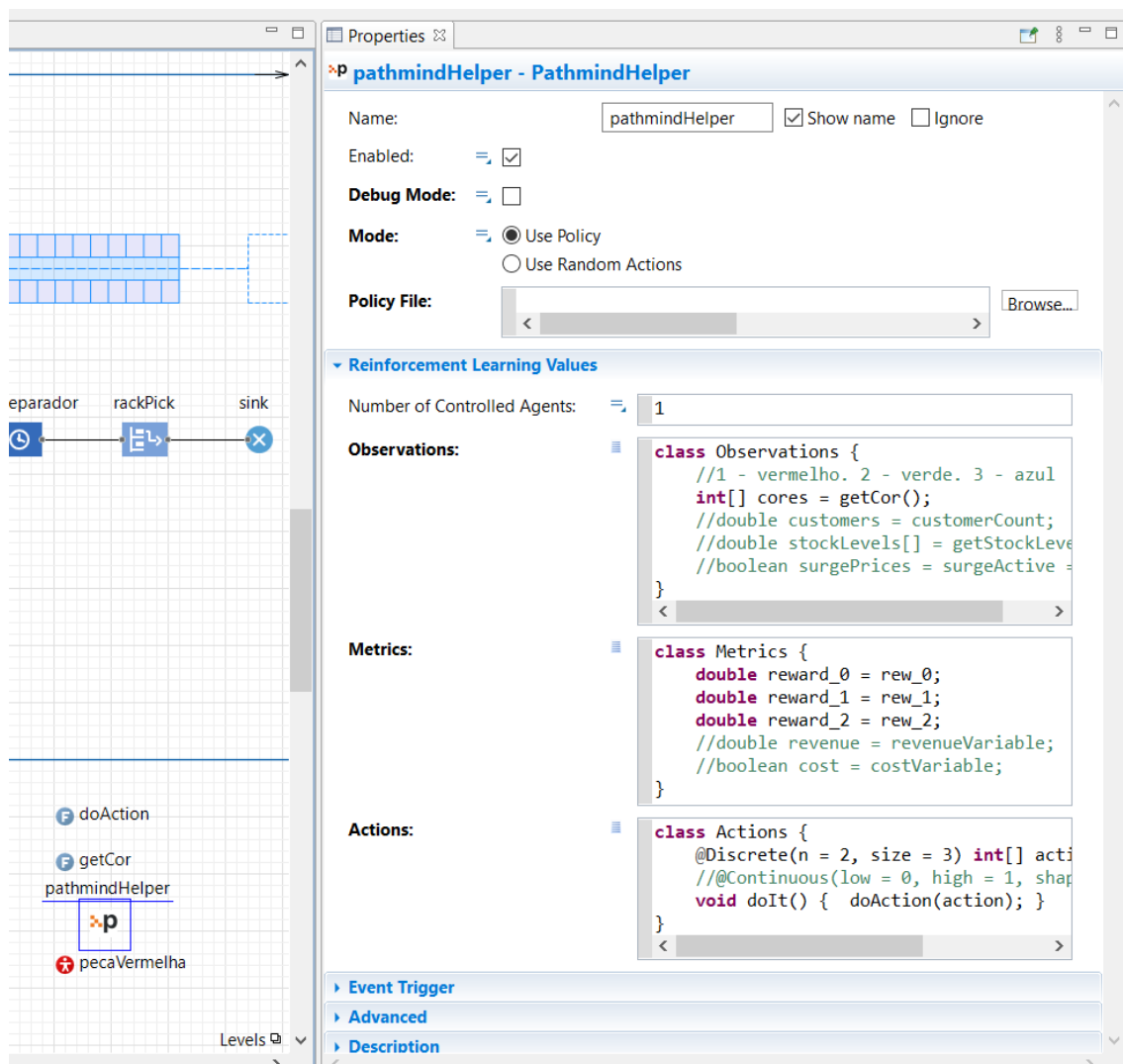
- Observações: Recebem as observações feitas pelo ambiente, no caso desse trabalho, as observações são quais os tipos de peças (vermelha, verde ou azul) e a partir dessas observações são definidas as funções de recompensa do projeto;
- Métricas: Definem as variáveis de recompensa. Para esse trabalho, foram utilizadas três variáveis de recompensa, uma para cada tipo de peça, sendo assim, elas são atualizadas de acordo com a observação feita. Caso a Inteligência Artificial acerte o tipo da peça, uma recompensa é dada para a variável referente a esse tipo, caso contrário, a recompensa é retirada;
- Ações: Definem a quantidade de possíveis alternativas para os estados do agente e definem também a quantidade de agentes utilizados no modelo. Nesse trabalho foram utilizados três tipos de agentes (peças vermelhas, verdes e azuis) e cada agente pode assumir dois estados (ser ou não ser daquela cor). As Ações também permitem definir quais ações devem ser tomadas para os agentes e as funções de recompensas. O exemplo utilizado aqui determina que se o tipo da peça for o mesmo que a IA indicou, soma-se o valor de um na recompensa e caso contrário, esse mesmo valor é subtraído.

A lógica para a utilização da IA no modelo apresentado foi a de que uma peça entra na fábrica e sua cor é lida e armazenada em um vetor de três posições, esse vetor é comparado com um vetor gerado pelo *PathMind Helper*® e baseado na comparação, as recompensas são calculadas.

Até a versão 8.7.3 do *AnyLogic*®, os últimos passos eram exportar o modelo para uma “*Java Standalone Application*”, criar uma função de recompensa e treinar a Inteligência Artificial para a criação de um *Policy* que, então, era inserido no modelo de forma que a simulação pudesse ser utilizada com a inteligência artificial. Nesse ponto se encontrava uma limitação de versão do *AnyLogic*®, pois exportar esse modelo para o formato necessário só é possível na versão Profissional do software, sendo assim, seria necessário adquirir essa licença para poder utilizar o *PathMind Helper*®.

É possível verificar a interface do *PathMind Helper*® através da Figura 4, na qual pode se observar a presença das suas três funções principais, observações, métricas e ações, bem como também a utilização do *Policy*, mas vale destacar que a pasta de origem do arquivo foi removida por fins de privacidade dos autores. Na Figura 4 também pode ser visto as funções que foram criadas, como “doAction” e “getCor”.

Figura 4 – Interface do *PathMind Helper*®



Fonte: Autores

5. Resultados

Como definido anteriormente, apenas a partir da versão 8.7.4 do *AnyLogic*® é possível utilizar a ferramenta *PathMind*® na versão PLE do software de simulação e, portanto, essa é a versão utilizada para os testes. A versão atual que o *PathMind*® se encontra é a versão 1.5.0, porém nela é possível verificar uma limitação de no máximo 20.000 iterações até que o *PathMind*® acuse um erro de limite excedido. Essa limitação não é encontrada na versão 1.3.4 e por isso essa foi a versão utilizada para os testes, visto que seriam necessários a criação de mais de 20.000 peças para os testes.

De forma a definir os resultados, a simulação foi feita em dois computadores diferentes, com 50 replicações em cada (totalizando 100 replicações) com cenários de 12 horas, 24 horas e 36 horas cada, primeiramente sem a utilização do *Policy* criado pelo *PathMind*® (com valores aleatórios gerados pela ferramenta) e posteriormente, após o treinamento da IA, utilizando o resultado gerado pelo *PathMind*®.

Para o teste do modelo foram utilizadas 3 peças que poderiam ser vermelhas, verdes ou azuis e um vetor era criado de acordo com cada uma das cores (vermelho: [1, 0, 0]; verde: [0, 1, 0]; azul: [0, 0, 1]). Sendo assim, o *PathMind*® tenta automaticamente adivinhar utilizando também um vetor de 3 posições, porém há a possibilidade dessa ferramenta criar vetores com mais de um valor 1, levando a conclusão de que o *PathMind*® poderia criar 8 vetores diferentes para tentar adivinhar o valor correto da cor da caixa. Dessa forma a quantidade de acertos esperada para a ferramenta deveria ser o total de peças criadas dividido por oito, portanto a taxa de acerto esperada para o *PathMind*® antes do treinamento é de 12,5%, o que se confirmou com os resultados dos testes.

Para o teste realizado antes do treinamento do IA com uma simulação de 12 horas, foram criadas 14.400 peças e a ferramenta *PathMind*® acertou uma média de 1800,5 peças com um desvio padrão de 40,9. Para o teste com 24h, o *AnyLogic*® criou 28.800 peças e os acertos do *PathMind*® tiveram média de 3.598,6 com desvio padrão de 61,1. Por último, para 36h, foram geradas 43.200 peças e os acertos tiveram média de 5.404,9 e desvio padrão de 79,5. Vale ressaltar que todas as médias ficaram próximas dos valores esperados que eram o total das caixas dividido por 8, a saber: para 12h a média seria 1.800, para 24h a média seria 3.600 e para 36h a média seria de 5.400, conforme mostrado na Tabela 1.

Tabela 1 – Taxa de acertos antes do treinamento

Antes do Treinamento					
Período (em horas)	Total Produzido	Média de Acertos	Média Esperada	Acertos (%)	Desvio Padrão
12	14.400	1.800,5	1.800	12,50%	40,9
24	28.800	3.598,6	3.600	12,50%	61,1
36	43.200	5.404,9	5.400	12,51%	79,5

Fonte: Autores.

Após o treinamento da Inteligência Artificial e a utilização do *Policy* do *PathMind*® no *AnyLogic*®, a taxa de acerto foi de 100% para todos os testes realizados. Após o treinamento, assim como antes do treinamento, foram coletadas, por cada uma das duas máquinas utilizadas, 50 amostras para cada período de tempo. Os tempos da simulação também foram os mesmos (12h, 24h e 36h) e todas as 300 amostras obtiveram o mesmo resultado de 100% de acerto. A princípio foi verificado que o *PathMind*® sempre errava apenas a cor de uma peça, porém foi descoberto que isso ocorria devido ao fato da simulação terminar antes da avaliação da ferramenta e ao se aumentar os tempos da simulação em 2 segundos, a ferramenta passou a acertar todas as peças.

De forma a verificar a capacidade da ferramenta *Pathmind*® foi realizado também um teste com uma simulação por um ano. No total foram criadas 10.512.000 peças (mais de dez milhões) e a ferramenta conseguiu identificar corretamente 10.511.882 peças, nessa simulação a ferramenta obteve uma pequena taxa de erro, sendo essa de 0,001122%.

6. Conclusões

O objetivo desse trabalho é a inclusão de ferramentas de IA na simulação baseada em agentes e utilizando o *PathMind Helper*® foi possível realizá-la de forma prática sem a necessidade de muitos conhecimentos em aprendizagem de máquina, sendo necessário que o usuário apenas entenda como seu modelo deve funcionar e quais parâmetros serão utilizados para definir as funções de recompensa.

A inclusão de elementos envolvendo inteligência artificial parece ser uma evolução natural para os modelos de simulação, sejam eles orientados a eventos discretos, baseados em agentes ou híbridos. Observa-se então um esforço das ferramentas de simulação e de aprendizado de máquina em promover integrações. Uma dessas iniciativas é a integração entre *Anylogic*® e *PathMind*® que se tornou possível para a versão PLE a partir da versão 8.7.4 do *AnyLogic*®.

O *PathMind Helper*® se mostrou uma ferramenta eficaz, pois os testes realizados obtiveram como resposta uma taxa de acerto de 100%, ou seja, foram identificadas corretamente todas as peças. Tais características a torna atrativa para profissionais e pesquisadores especializados em simulação uma vez que essa ferramenta pode ser aplicada em problemas de logística, separação e identificação de materiais em geral, por exemplo.

Outras vantagens da ferramenta, além da facilidade de uso e taxa de acerto é o fato de que todo o treinamento acontece na nuvem, isso permite que os modelos sejam exportados para treinamento e caso ocorram quedas de energia ou problemas com a máquina onde se encontra o modelo, o treinamento não é perdido e após a conclusão um e-mail ainda é enviado para informar que o *Policy* está pronto.

Para trabalhos futuros sugere-se a utilização do *PathMind*® em modelos reais de forma a verificar se, em modelos maiores, essa taxa de acerto se mantém tão alta quanto no modelo criado para esse trabalho.

Agradecimentos

Os autores desse trabalho agradecem à FAPEMIG, à CAPES e ao CNPq pelo apoio financeiro no decorrer dessa pesquisa. Agradecemos também ao Edward Junprung e toda a equipe da *PathMind Helper*® que não mediram esforços para nos auxiliar na resolução dos problemas durante esse trabalho.

Referências

- Baines, T. S.; Mason, S.; Siebers, P. O.; Ladbroke, J. Humans: the missing link in manufacturing simulation? **Simulation Modelling Practice and Theory**, 12: 515–526, 2004.
- Borshchev, A.; Filippov, A. From system dynamics and discrete event to practical agent based modeling: reasons, techniques, tools. In: **Proceedings of the 22nd international conference of the system dynamics society**. 2004.
- Brailsford, Sally C. et al. Hybrid simulation modelling in operational research: A state-of-the-art review. **European Journal of Operational Research**, v. 278, n. 3, p. 721-737, 2019.
- Chwif, L.; Medina, A. C. **Modelagem e Simulação de Eventos Discretos: Teoria e Aplicações**. São Paulo: Ed. dos Autores, 2015.

Deist, Timo M. et al. Simulation-assisted machine learning. **Bioinformatics**, v. 35, n. 20, p. 4072-4080, 2019.

Dubiel, B.; Tsimhoni, O. Integrating agent based modeling into a discrete event simulation. In: **Proceedings of the Winter Simulation Conference, 2005**. IEEE, p. 9, 2005.

Ferreira, W. De P.; Armellini, Fabiano; De Santa-Eulalia, Luis Antonio. Simulation in industry 4.0: A state-of-the-art review. *Computers & Industrial Engineering*, p. 106868, 2020.

Ingalls, R. G.; Introduction to Simulation. In: **Winter Simulation Conference, Proceedings...** Miami, FL, USA, 2008.

Kaelbling, Leslie Pack; Littman, Michael L.; Moore, Andrew W. Reinforcement learning: A survey. **Journal of artificial intelligence research**, v. 4, p. 237-285, 1996.

Kober, Jens; Bagnell, J. Andrew; Peters, Jan. Reinforcement learning in robotics: A survey. **The International Journal of Robotics Research**, v. 32, n. 11, p. 1238-1274, 2013.

Liao, Yongxin et al. Past, present and future of Industry 4.0-a systematic literature review and research agenda proposal. *International journal of production research*, v. 55, n. 12, p. 3609-3629, 2017.

Macal, Charles M. Everything you need to know about agent-based modelling and simulation. *Journal of Simulation* 10.2 (2016): 144-156.

Macal, C. M.; North, M. J. Tutorial on agent-based modeling and simulation. In: **Proceedings of the 37th conference on Winter simulation**. p. 2-15, 2005.

Mavrovouniotis, Michael (Ed.). **Artificial intelligence in process engineering**. Elsevier, 2012.

Mccarthy, John. **What is artificial intelligence?** 1998.

Melão, N., Pidd, M. **Using component technology to develop a simulation library for business process modeling**. *European Journal of Operational Research*, 172: 163–178, 2006.

Mitchell, R. S.; Michalski, J. G.; Carbonell, T. M. **An artificial intelligence approach**. Berlin: Springer, 2013.

Mladeníć, Dunja et al. Using machine learning techniques to interpret results from discrete event simulation. In: **European Conference on Machine Learning**. Springer, Berlin, Heidelberg, 1994. p. 399-402.

Montevecchi, J. A. B., Pinho, A. F. DE, Leal, F, Marins, F. A. S. Application of design of experiments on the simulation of a process in an automotive industry. In: **Proceedings of the Winter Simulation Conference**, Washington, DC, USA, 2007.

Montevecchi, José Arnaldo Barra et al. Conceptual modeling in simulation projects by mean adapted IDEF: an application in a Brazilian tech company. In: *Proceedings of the 2010 winter simulation conference*. IEEE, 2010. p. 1624-1635.

Nilsson, Nils J. **Principles of artificial intelligence**. Morgan Kaufmann, 2014.

PathMind Helper, Inc., **Pathmind Technology** Disponível em: <<https://www.anylogic.com/features/artificial-intelligence/pathmind/>>, 2020. Acesso em 14/12/2020.

Poddar, T. et al. Digital twin bridging intelligence among man, machine and environment. In: **Offshore Technology Conference Asia**. Offshore Technology Conference, 2018.

Qiang, Wang; Zhongli, Zhan. Reinforcement learning model, algorithms and its application. In: **2011 International Conference on Mechatronic Science, Electric Engineering and Computer (MEC)**. IEEE, 2011. p. 1143-1146.

Schriber, T. J.; Brunner, D. T. Inside discrete-event simulation software: how it works and why it matters. In: **Winter Simulation Conference, Proceedings...** Huntington Beach, CA, USA, 2011.

Siebers, P.O.; Macal, C.M.; Garnett, J.; Buston, D.; Pidd, M. et al. Discrete-event simulation is dead, long live agent-based simulation!. **Journal of Simulation**, v. 4, n. 3, p. 204-210, 2010.

Sobie, Cameron; Freitas, Carina; Nicolai, Mike. Simulation-driven machine learning: Bearing fault classification. **Mechanical Systems and Signal Processing**, v. 99, p. 403-419, 2018.

Standridge, C. R. Teaching simulation using case studies. In: **2000 Winter Simulation Conference Proceedings (Cat. No. 00CH37165)**. IEEE, 2000. p. 1630-1634.

Sutton, Richard S.; Barto, Andrew G. **Reinforcement learning: An introduction**. MIT press, 2018.

Swinerd, Chris; Mcnaught, Ken R. Design classes for hybrid simulations involving agent-based and system dynamics models. **Simulation Modelling Practice and Theory**, v. 25, p. 118-133, 2012.

The AnyLogic Company, **Boost Performance Over Your Baseline With Pathmind AI**. Disponível em: <<https://www.anylogic.com/features/artificial-intelligence/pathmind/>>, 2020. Acesso em 14/12/2020

Uhlemann, Thomas H.-J.; Lehmann, Christian; Steinhilper, Rolf. The digital twin: Realizing the cyber-physical production system for industry 4.0. **Procedia Cirp**, v. 61, p. 335-340, 2017.

Xu, Li Da; XU, Eric L.; LI, Ling. Industry 4.0: state of the art and future trends. **International Journal of Production Research**, v. 56, n. 8, p. 2941-2962, 2018.