



GRADUS
EDITORA

Barbara Frigini De Marchi
Daniela Dadalto A. Missawa
Claudia Broetto Rossetti
(Organizadoras)

Temas Atuais em **DESENVOLVIMENTO HUMANO**



Temas atuais em desenvolvimento humano, DE MARCHI, Barbara Frigini; MISSAWA, Daniela Dadalto Ambrozine; ROSSETTI, Claudia Broetto (Orgs). Gradus Editora, 2022. 276p.. : il. (algumas color.); PDF.

FICHA TÉCNICA

Editor-chefe

Lucas Almeida Dias

Projeto gráfico

Paulo Ricardo Cavalcante da Silva

Diagramação e Projeto gráfico

Natália Huang Azevedo Hypólito

Revisão

Lucas Almeida Dias

Comitê Editorial Científico – Gradus Editora 2021/2022

Dra. Ana Beatriz Duarte Vieira

Dra. Ana Cláudia Bortolozzi

Dra. Andreia de Bem Machado

Dr. Carlos Gomes de Castro

Dra. Cintya de Oliveira Souza

Dra. Daniela Marques Saccaro

Dr. Douglas Manoel Antonio de Abreu Pestana Dos Santos

Dr. Fábio Roger Vasconcelos

Dra. Gladys del Carmen Medina Morales

Dr. Gustavo Schmitt

Dra. Janaína Muniz Picolo

Dr. Leandro Antônio dos Santos

Dr. Luís Rafael Araújo Corrêa

Dra. Manuela Costa Melo

Dra. Márcia Lopes Reis

Dra. Renata Cristina Lopes Andrade

Dr. Thiago Henrique Omena

Dr. Yan Corrêa Rodrigues

Dados Internacionais de Catalogação na Publicação (CIP)

Temas atuais em desenvolvimento humano, DE MARCHI, Barbara Frigini; MISSAWA, Daniela Dadalto Ambrozine; ROSSETTI, Claudia Broetto (Orgs). Gradus Editora, 2022. 276p.. : il. (algumas color.); PDF.

Inclui bibliografias.

978-65-88496-80-0

1. Psicologia 2. Desenvolvimento Humano 3. Pesquisas

CDD 370.00



SUMÁRIO

Prefácio _____ 3

Maria Thereza Costa Coelho de Souza

Capítulo 1. Tutorial de modelagem baseada em agentes para pesquisas em desenvolvimento humano _____ 11

Hugo Cristo Sant'Anna

Capítulo 2. Criança pode escolher? Normatividade e desenvolvimento moral _____ 47

Thaís Síndice Fazenda Coelho; Betânia Alves Veiga Dell'Agli; Luciana Maria Caetano

Capítulo 3. O cuidado na atenção psicossocial a crianças e adolescentes que fazem uso de substâncias psicoativas: uma análise a partir da teoria de Jung _____ 69

Kelly Guimarães Tristão; Luziane Zacché Avellar

Capítulo 4. A Teoria do amadurecimento humano de Winnicott: pressupostos teóricos e suas contribuições para se pensar sobre o acolhimento institucional de crianças e adolescentes _____ 99

Ivy Campista Campanha-Araujo; Luziane Zacché Avellar

Capítulo 5. Um estudo sobre o perdão em crianças e adolescentes na perspectiva da Psicologia do Desenvolvimento _____ 125

Caroline Benezath Rodrigues Bastos; Sávio Silveira de Queiroz

Capítulo 6. O lugar da violência nas pesquisas com jogos eletrônicos _____ **143**

Eduardo Silva Miranda; Claudia Broetto Rossetti

Capítulo 7. Contribuições da etnografia de Fine para os estudos do role-playing game (RPG) de mesa _____ **163**

Ana Carolina Martins Braga; Eduardo Silva Miranda; Marcelo de Almeida Ferreri; Claudia Broetto Rossetti

Capítulo 8. Atividades parentais: a dimensão do cuidado na parentalidade de casais do mesmo sexo por via da adoção _____ **183**

Carolina Monteiro Biasutti; Mariana Calhau de Figueiredo; Célia Regina Rangel Nascimento

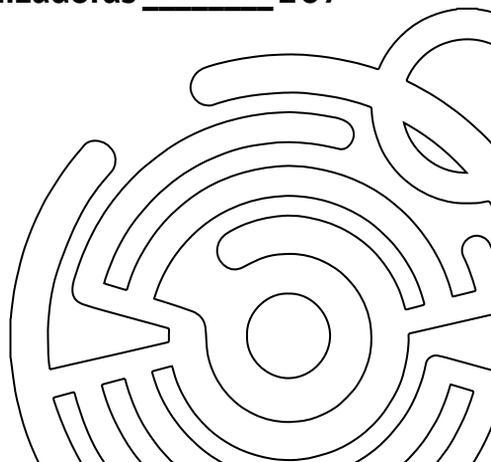
Capítulo 9. Parentalidade após a separação: impactos e desafios da guarda paterna _____ **213**

Carolina Corrêa Vilaça Miceli; Célia Regina Rangel Nascimento

Capítulo 10. O que atravessa uma mulher negra e idosa? _____ **247**

Jamylle Adrienne de Lima Silva; Maria da Conceição Florencio Monteiro Bezerra; Claudimara Chisté Santos; Suely Emilia de Barros Santos

Currículos dos Autores e Organizadoras _____ **267**





CAPÍTULO 1.

Tutorial de modelagem baseada em agentes para pesquisas em desenvolvimento humano

Hugo Cristo Sant'Anna

Introdução

Este capítulo apresenta aplicações da modelagem baseada em agentes (*Agent-Based Modeling*, ABM) a pesquisas do desenvolvimento humano em formato tutorial. As seções a seguir foram organizadas visando oferecer panoramas breves sobre as origens, conceitos, principais métodos e técnicas de modelagem em geral, e ABM em especial, apoiando a análise de um modelo computacional elaborado para comparar o desempenho de agentes operando segundo as teorias de Jean Piaget (1896-1980) e Lev Vygotsky (1896-1934). Este modelo foi implementado em NetLogo, ambiente para a realização de pesquisas em ABM, desenvolvido desde 1999 pelo Center for Connected Learning and Computer-Based Modeling da Northwestern University (EUA).

A presente exposição não pressupõe conhecimentos prévios de programação NetLogo, modelagem matemática ou das teorias de Piaget e Vygotsky. Termos técnicos e conceitos específicos serão introduzidos quando necessário, mantendo o texto acessível a pesquisadores iniciantes no tema. Ao final do capítulo, espera-se que o leitor seja capaz de vislumbrar possibilidades de aplicação de ABM e do ambiente NetLogo a seus próprios estudos do desenvolvimento humano.

Modelagem em Psicologia

Segundo Coombs, Dawes e Tversky (1970), ferramentas, métodos e modelos matemáticos são utilizados na Psicologia desde o século XIX. No entanto, a modelagem computacional se difundiu apenas após o aumento da disponibilidade de computadores, a partir dos anos 1950. Hovland (1960) analisou estudos sobre resolução de problemas, teorias de aprendizagem, formação de conceitos, plausibilidade de redes neurais artificiais, modelos de memória e esquecimento, defendendo a aplicação de simulações desses processos em computadores como estratégia para aumentar a compreensão sobre o funcionamento cognitivo de seres humanos. Abelson (1968), no volume de métodos da segunda edição do *Handbook of Social Psychology*, revisou estudos sobre processos cognitivos e comportamento social que empregaram simulações computacionais. A obra clássica de Newell e Simon (1972) consiste em tratamento extenso sobre a modelagem de processos cognitivos de seres humanos e resolução de problemas utilizando computadores, sistematizando mais de duas décadas de pesquisas. Os periódicos *The Journal of Mathematical Psychology*¹ e *Cognitive Science*², cujas edições têm presença constante de estudos de modelagem, foram fundados respectivamente em 1964 e 1976, enquanto a *Society for Mathematical Psychology* e *Cognitive Science Society*, sociedades vinculadas aos periódicos citados, foram fundadas em 1977 e 1979.

A popularização dos computadores, a partir dos anos 1980, parece ter sido o evento que contribuiu de forma definitiva

1 Disponível em <<https://mathpsych.org/page/history>>. Acesso em: 05 mar 2022.

2 Disponível em <<https://cognitivesciencesociety.org/about/>>. Acesso em: 05 mar 2022.





para a ampliação do uso das técnicas de modelagem entre pesquisadores da Psicologia (VALLACHER; NOWAK; READ, 2017). Nessa época, Ostrom (1988) relatou que o preconceito de psicólogos quanto aos usos de simulações e modelos computacionais havia perdido força, mas persistia em razão de quatro fontes: 1) falta de conhecimentos especializados sobre programação na formação; 2) falta de conhecimento sobre avanços teóricos obtidos por meio de simulações; 3) poucos exemplos de pesquisadores de sucesso adeptos da modelagem computacional, para inspirar as novas gerações; e 4) persistência da percepção de que métodos computacionais seriam específicos para estudos em Psicologia Cognitiva ou restritos a experimentos controlados de laboratório. Há convergências entre estas razões e a avaliação de Abelson (1968), realizada duas décadas antes, principalmente quanto à falta de oportunidades de formação específica e às aplicações restritas de computadores para estudos em Psicologia. Não obstante, a década de 1980 foi marcada pelo surgimento e aperfeiçoamento de muitos modelos computacionais da cognição (cf. THAGARD, 1998).

Mais de trinta anos depois, Iris van Rooij (2022) argumentou que o uso de modelos em Psicologia ainda é raro, e que haveria consenso crescente de que tal negligência estaria atrasando progressos na área. A autora também citou a falta de oportunidades de formação específica na temática em currículos de Psicologia, e apontou uma segunda razão para a falta de modelagem: a dominância de práticas empiristas ingênuas na área, que privilegiam observação e testes empíricos em detrimento de compromissos teóricos. Por fim, há inúmeros tutoriais e revisões recentes elaboradas com o intuito de disseminar práticas de modelagem entre pesquisadores da Psicologia, problematizando questões

ontológicas e epistemológicas de modelos (cf. SMALDINO, 2020; VAN ROOIJ; BLOKPOEL, 2020; GUEST; MARTIN, 2021).

O ambiente NetLogo

NetLogo (WILENSKY, 1999) é uma linguagem de programação multiagentes e ambiente de modelagem para a simulação de fenômenos complexos, sejam naturais ou sociais. É gratuita e está disponível para a maioria dos sistemas operacionais atuais³. Sua distribuição acompanha documentação detalhada, tutoriais e biblioteca com exemplos de modelos de fenômenos das áreas de Biologia, Medicina, Física, Química, Matemática, Ciência da Computação, Economia e Psicologia (TISUE; WILENSKY, 2004). Desde 2015, NetLogo pode ser utilizado na versão Web⁴, acessível por navegadores, com poucas diferenças de recursos em relação à versão instalável em computadores. O ambiente NetLogo tem três abas (Figura 1-1), nas quais o pesquisador passa a maior parte do processo de modelagem: *Interface*, *Informação* e *Código*⁵.

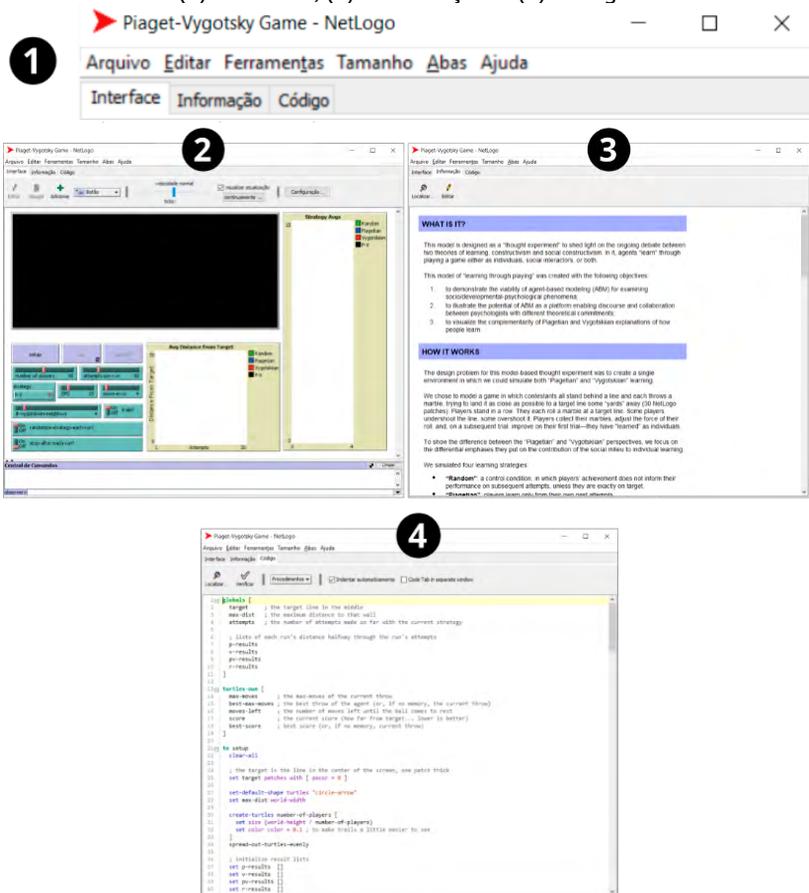
3 NetLogo requer o ambiente de execução Java (JRE) da Oracle, disponível para sistemas operacionais Windows, macOS e variantes de Linux em <<https://java.com/pt-BR/>>. Acesso em: 05 mar. 2022.

4 Disponível em <<https://www.netlogoweb.org>>. Acesso em: 05 mar. 2022.

5 A tradução para português do Brasil acompanha a distribuição básica e pode ser habilitada durante a instalação ou nas preferências do ambiente – menu superior, opção *Tools*, item *Preferences*.



Figura 1 – Abas (1) do ambiente NetLogo: (2) Interface, (3) Informação e (4) Código.



Fonte: produzido pelo autor, exibindo modelo de Abrahamson e Wilensky (2005a).

A execução e inspeção do modelo ocorrem na aba Interface (Figura 1-2). Esta exibe a saída gráfica, denominada *mundo*, itens de controle (botões, deslizadores, interruptores, selecionadores, campo para entrada de dados) e monitoramento (gráficos e campos para saída de dados) do modelo. A aba

Informação (Figura 1-3) apresenta detalhes sobre o modelo, fornecidas por seus autores para auxiliar outros pesquisadores a entenderem o seu funcionamento. Por padrão, as seções desta aba são: o que é, como funciona, como utilizar, questões a observar, questões a tentar, particularidades NetLogo utilizadas pelo modelo, modelos relacionados, créditos e referências. Nem todas as seções da aba Informações precisam ser preenchidas pelos autores do modelo, embora seja parte do conjunto de boas práticas de NetLogo. A terceira aba corresponde à listagem do código NetLogo (Figura 1-4), construído a partir dos elementos primitivos da linguagem⁶ e procedimentos criados pelo usuário.

Em termos práticos, modelos criados em NetLogo empregam três tipos de entidades: 1) tartarugas (*turtles*), agentes móveis que interagem entre si e com o entorno; 2) retalhos (*patches*) do terreno, áreas quadradas fixas do mundo que interagem entre si e com as tartarugas; 3) o observador (*observer*) único, que dá comandos e observa eventos e entidades no ambiente. Durante seu deslocamento no mundo, tartarugas podem “sentir” e “responder” às propriedades de outras tartarugas e retalhos. Programas escritos em NetLogo utilizam essas relações para modelar os fenômenos de interesse.

Segundo seus desenvolvedores, um dos objetivos da concepção de NetLogo foi a *reprodutibilidade* (TISUE; WILENSKY, 2004): modelos criados no ambiente operam de modo determinístico, isto é, sua execução seguirá os mesmos passos a atingirá os mesmos resultados em todos os computadores. Tal condição permite a pesquisadores testarem seus modelos

6 O dicionário da documentação organiza e apresenta exemplos de uso de todos os elementos primitivos de NetLogo. Disponível em <<https://ccl.northwestern.edu/netlogo/bind/>>. Acesso em: 05 mar. 2022.





milhares de vezes, variando suas configurações iniciais (parâmetros) e avaliando confiavelmente as consequências dessas variações. O recurso de NetLogo denominado *Behavior Space* é responsável por gerenciar as séries de execuções com variações de parâmetros, armazenando os resultados em planilhas e tabelas para análise posterior.

O modelo NetLogo Piaget-Vygotsky Game

O trabalho que introduziu o modelo foi apresentado e publicado nos anais da 35ª reunião anual da Jean Piaget Society, em 2005. Dor Abrahamson, então bolsista de pós-doutorado do Center for Connected Learning and Computer-Based Modeling (CCL) da Northwestern University (EUA), e Uri Wilensky, coordenador do CCL e supervisor de Abrahamson, são respectivamente o primeiro e segundo autores do artigo e do modelo.

No artigo (ABRAHAMSON; WILENSKY, 2005b), os autores relataram que as motivações para a construção do modelo surgiram em uma conferência internacional em que um pesquisador (não identificado) teria sido criticado por endossar simultaneamente o construtivismo associado a Jean Piaget (1896-1980) e socioconstrutivismo associado a Lev Vygotsky (1896-1934) como abordagens para o estudo da aprendizagem. As rivalidades entre as abordagens, a despeito dos esforços de aproximação (cf. DE LA TAILLE; DE OLIVEIRA; DANTAS, 1992; COLE; WERTSCH, 1996), persistiriam em razão da falta de um “campo comum”. Abrahamson e Wilensky alegaram que seguidores das abordagens de Piaget e Vygotsky encontrariam dificuldades para estabelecer diálogos produtivos por adotarem métodos, perspectivas e terminologias distintas. O modelo computacional

(ABRAHAMSON; WILENSKY, 2005a), intitulado “*I’m Game!*” e referido no restante do presente capítulo como “modelo P-V”, consistiu na tentativa de transpor o diálogo entre as duas abordagens para o “campo comum” fornecido pela modelagem baseada em agentes (ABM), expressando aspectos teóricos de ambas na linguagem de programação NetLogo.

Os objetivos do modelo P-V declarados no artigo foram: 1) demonstrar a viabilidade de aplicação de ABM ao exame de fenômenos psicológicos, sociais e do desenvolvimento; 2) ilustrar o potencial de ABM como plataforma para fomentar o diálogo e colaboração entre psicólogos com filiações teóricas distintas; 3) visualizar a complementariedade das explicações de Piaget e Vygotsky quanto a como as pessoas aprendem. Houve esforço para que a simulação fosse simples e concisa, todavia mantendo as distinções significativas entre as teorias.

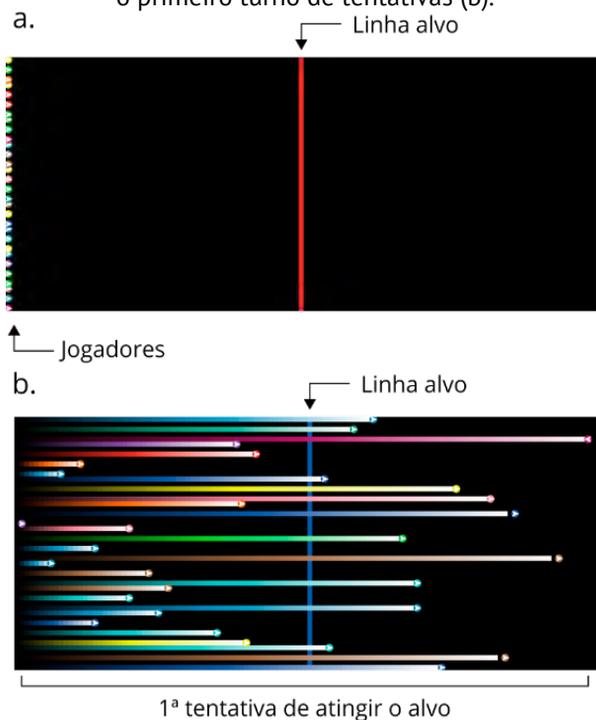
Abrahamson e Wilensky (2005a) escolheram brincadeiras de crianças em grupo para modelar a aprendizagem humana em contextos sociais, com a justificativa de que Piaget e Vygotsky estudaram relações entre brincadeiras e desenvolvimento cognitivo em suas obras. Neste contexto, o grupo de crianças estaria engajado em atividades lúdicas *explícitas* (jogando “bolas de gude”), permitindo aos modeladores definirem, sem equívocos, os objetivos e habilidades subjacentes necessárias às crianças para se engajarem na atividade. Estes jogos, de acordo com os autores, são comumente discutidos nos termos de procedimentos, podendo ser descritos em programas de computador como regras a serem seguidas pelos jogadores.

A construção do modelo prosseguiu pela criação de um ambiente único em que a aprendizagem segundo pressupostos piagetianos e vygotskyanos pudesse ser simulada, enfatizando os diferentes entendimentos de cada teoria sobre as



contribuições do meio social para a aprendizagem individual. O jogo modelado consiste em jogadores em fila à esquerda do mundo de NetLogo (Figura 2-a), tentando, em turnos, atingir a linha situada na parte central da tela com lançamentos de bolas de gude (Figura 2-b).

Figura 2 – Espaço simulado do jogo (a) e o primeiro turno de tentativas (b).



Fonte: adaptado de Abrahamson e Wilensky (2005a).

No jogo, buscou-se mensurar e visualizar progressos incrementais no desempenho dos aprendizes simulados em quatro⁷ condições, rotuladas por Abrahamson e Wilensky como 7 No artigo (ibid., p. 17), os autores contaram duas condições, citando ainda a piagetiana-vygotskyana e aleatória (controle). Neste texto, optou-

“caricaturas grosseiras” das teorias: 1) *piagetiana* – jogadores interagem com os objetos no ambiente, acomodando seus esquemas baseados em *feedbacks*; b) *vygotskyana* – jogadores aprendem por imitação, participando de uma atividade organizada que inclui outros jogadores mais experientes no desempenho da habilidade alvo; c) *piagetiana-vygotskyana* – jogadores aprendem tanto com o próprio desempenho quanto com o desempenho dos outros; e 4) controle (sem aprendizado, lançamentos aleatórios).

A expectativa dos autores foi que as tipificações “caricaturais” dos processos de aprendizagem de Piaget e Vygotsky seriam suficientes para descrever regras simples para os aprendizes individuais simulados e mesmo assim gerar padrões comportamentais mais interessantes e reveladores no nível grupal. Como será discutido nas seções subsequentes, esta é uma das premissas centrais da modelagem de sistemas complexos, relacionada à articulação de fenômenos nos níveis *micro* e *macro* e ao conceito de *emergência*.

Ainda na Figura 2-b, pode-se inspecionar os desempenhos individual e coletivo dos aprendizes na primeira tentativa de atingir a linha alvo na simulação. Combinando as condições modeladas e o resultado de um turno da simulação, a dinâmica decorrente das regras do jogo torna-se mais evidente. A cada turno, os jogadores lançam bolas de gude a posições situadas antes, depois ou sobre a linha alvo, avaliam este resultado e fazem ajustes no lançamento seguinte, tentando chegar mais perto ou atingir o objetivo.

A variação entre as condições do modelo altera o procedimento NetLogo de ajuste feito pelo jogador para o

se por citá-las como quatro condições para evitar confusões.





próximo turno. Abrahamson e Wilensky (2005b, p. 28) citaram vagamente, nas conclusões do artigo, os pressupostos da teoria piagetiana que teriam sido implementados na respectiva condição do modelo – esquemas, assimilação, acomodação, especialização e equilíbrio. Piaget (1964/2010) atribuiu importante valor explicativo à noção de equilíbrio em Psicologia, caracterizando-a como propriedade intrínseca e constitutiva da vida orgânica e mental. A equilíbrio é a coordenação necessária das relações entre os outros três fatores do desenvolvimento humano (maturação, interações com o meio físico e com o meio social), tendendo a assegurar o equilíbrio móvel e ativo entre fatores internos e externos, ou entre as funções de assimilação e acomodação.

A capacidade cognitiva de antecipar ativamente as perturbações externas e suas respectivas compensações, por operações recíprocas ou inversas (PIAGET, 1964/2010b), nos parece relevante para compreender a implementação do modelo P-V em NetLogo. No procedimento identificado por *p-adjust*⁸, o jogador avalia sozinho sua última jogada quanto à distância da linha alvo, comparando-a com a sua *melhor* jogada. Este valor é registrado pela variável nomeada *best-score* do código e integra o conjunto de propriedades de todos os jogadores do modelo. Sempre que a posição alcançada pela última jogada, registrada pela variável *score*, for menor que *best-score*, o procedimento *p-adjust* do modelo atribui o valor de *score* a *best-score*. Valores menores consistem em desempenho superior porque significam a redução da distância entre a linha alvo e a posição atingida pelo lançamento da bola de gude. As variáveis *max-moves* e

⁸ O procedimento *p-adjust* pode ser visualizado na aba Código, linhas 177 a 183.

best-max-moves registram a distância atingida pelo último lançamento e a distância atingida pelo *melhor* lançamento. Do mesmo modo que *best-score*, a variável *best-max-moves* é atualizada pelo procedimento *p-adjust* sempre que houver progressos no desempenho do jogador.

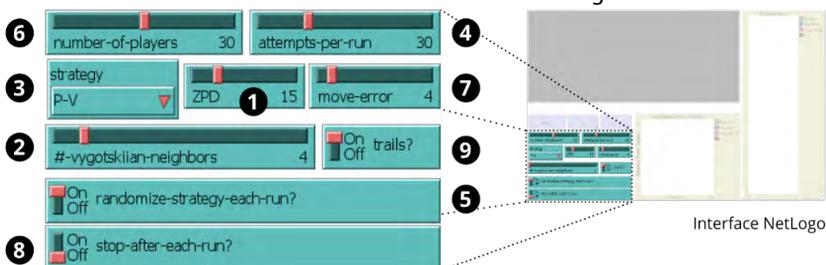
Voltando à teoria piagetiana, a interpretação de Abrahamson e Wilensy acerca do processo de equilíbrio seria melhor explicada considerando a ocorrência do jogo no mundo real. A criança não inicia o jogo desprovida de conhecimentos anteriores sobre os objetos, regras e ações pertinentes ao contexto. Conhece bolas de gude, suas dimensões, peso, a sensação de segurá-las nas mãos ou de lançá-las; entende as noções de atingir um alvo, de errá-lo por muito ou por pouco, e assim por diante. A declaração das regras e objetivos do jogo demanda novas coordenações entre os esquemas de ação a cada tentativa, ajustando a força necessária para lançar a bola de gude em direção ao alvo. No jogo real, a criança buscaria o balanço entre o peso do objeto, movimentos dos braços e mãos, distância a ser atingida e condições ambientais. No modelo P-V, essa dinâmica foi bastante simplificada, desconsiderando-se fatores físicos (p.ex. peso da bola de gude, gravidade, atrito do terreno) e simulando aspectos dos funcionamentos perceptivo e motor da criança pela atualização contínua das variáveis *best-score* e *best-max-moves*.

A condição vygotskyana, descrita pelo procedimento *v-adjust*⁹, incorpora o desempenho de jogadores vizinhos na avaliação da última jogada, simulando o funcionamento da zona de desenvolvimento proximal (ZDP). Este construto foi definido como a distância entre o nível de desenvolvimento real e o nível de desenvolvimento potencial, uma condição em 9 Linhas 185-201 do código.



que a criança consegue solucionar problemas que estão além de seu nível de desenvolvimento, sob a orientação de adultos ou em colaboração com outras crianças mais experientes (VIGOTSKY, 1984/2007). Sendo assim, nesta condição a criança simulada prestaria atenção às jogadas de outras crianças na brincadeira, avaliando o desempenho delas para imitá-las. Esta avaliação não foi modelada de modo fixo, podendo ser ajustada por dois controles na aba Interface (Figura 3).

Figura 3 – Detalhe dos controles da simulação do modelo P-V na interface NetLogo.



Fonte: adaptado de Abrahamson e Wilensky (2005a).

O deslizador ZPD (Figura 3-1) estabelece a diferença máxima entre o desempenho do jogador atual e o de outras crianças, sendo o limite inferior igual a zero (sem imitação). Por exemplo, se *score* da última tentativa do jogador atual for 30 e *score* de outra criança for 5, o deslizador com valor 15 não permitirá que o primeiro jogador imite o segundo, pois a diferença entre os desempenhos é igual a 25. Em termos teóricos, seria o mesmo que dizer que o jogador atual não consegue imitar o desempenho da outra criança, pois esta seria muito mais experiente e estaria além do nível potencial da primeira.

No código NetLogo, na hipótese de o desempenho superior de um dos vizinhos estar contido na ZPD do jogador atual,

este copiará *best-score* e *best-max-moves* daquele vizinho. No sentido inverso, valores altos para a ZDP permitem que jogadores com performances baixas possam rapidamente atingir o alvo observando os acertos de jogadores mais experientes. O deslizador nº2 da Figura 3 define o número de vizinhos observados e passíveis de imitação. Este controle tem valor mínimo igual a 2, revelando as escolhas de Abrahamson e Wilensky quanto ao que seria uma ZDP mínima: três jogadores, cada um observando as condutas dos outros dois. Cabe mencionar que a interpretação da teoria vygotskyana realizada pelo modelo P-V é indiferente a heteronomias no âmbito da ZDP e a dificuldades na transição das resoluções de problemas bem-sucedidas no nível interindividual para o funcionamento psicológico intraindividual (cf. WERTSCH, 1985, cap. 6; CASTORINA, 2010).

A terceira condição, piagetiana-vygotskyana, combina as descrições dos modelos anteriores no procedimento *pv-adjust*¹⁰. Primeiramente, o jogador avalia seu próprio desempenho em relação à tentativa anterior e, se for superior, atualiza a melhor jogada e distância da melhor jogada. Em seguida, avalia o desempenho do número de vizinhos dentro da ZDP e realiza ou não as atualizações devidas das variáveis, simulando complementaridades entre as abordagens construtivista e socioconstrutivista. A última condição é a de controle, *r-adjust*, que a cada jogada sorteia aleatoriamente a distância para lançar a bola de gude e só atualiza *best-score* e *best-max-moves* se acertar o alvo, sem registrar o desempenho dos turnos sem acertos.

Nas configurações de execução do modelo, pode-se definir qualquer uma das quatro condições por meio do selecionador 10 Linhas 203 a 222.





nº3 da Figura 3. Cada execução compreende o número de turnos definido pelo deslizador nº4, enquanto o interruptor nº5 ativa a seleção aleatória das estratégias de cada execução. Dessa forma, execuções com 30 turnos registram o desempenho dos jogadores na mesma condição durante 30 tentativas, para em seguida selecionar outra condição e realizar mais 30 jogadas. O deslizador nº6 permite alterar o número total de jogadores da simulação e o de nº7 controla a quantidade de ruído (erro) no processo. Abrahamson e Wilensky justificaram a inclusão deste controle para “tornar o processo de aprendizagem mais realista” (2005b, p. 20), uma vez que ele afeta o procedimento que calcula o próximo lançamento (*max-moves*). Os detalhes deste procedimento são irrelevantes no momento, bastando mencionar que a posição a ser atingida pelo próximo lançamento resulta da soma valores aleatórios normalmente distribuídos à distância registrada na melhor jogada, *best-max-moves*. Esses valores aleatórios são gerados em função do ajuste da quantidade de ruído, da melhor jogada (*best-score*), e da maior distância permitida para lançamentos, que é fixa¹¹. Por último, o estado do interruptor do controle nº8 na posição ligada (*on*) interromperá a execução da simulação ao fim do número de tentativas definidas pelo controle nº4. Já o interruptor sinalizado pelo nº9 na Figura 3 tem função estética, ativando ou desativando os “rastros” deixados pelos lançamentos a cada jogada.

O modelo P-V expressa as suposições teóricas dos autores acerca das abordagens construtivista e socioconstrutivista na forma de procedimentos NetLogo. Os autores mencionaram que o programa tem pouco mais de cinco páginas esparsas

¹¹ O procedimento foi implementado com diferenças entre as condições. Ver linhas 164, 170 e 173.

de programação, cerca de 820 palavras (4000 caracteres), das quais 455 consistem na parte central do código do modelo. Embora seja difícil para o leitor pouco familiarizado com linguagens de programação entender esses números, a sintaxe de NetLogo é bastante compacta quando comparada a outras linguagens utilizadas em modelagem computacional.

Considerando as publicações e revisões subseqüentes dos modelos, pode-se inferir que ao menos o objetivo de estabelecer o campo comum para a discussão das abordagens foi atingido. Em um trabalho subseqüente, Abrahamson, Wilensky e Levin (2007) expuseram desdobramentos da disseminação do modelo P-V para outros grupo de pesquisa, incluindo críticas sobre simplificações teóricas e decisões dos modeladores. No caso da condição vygotskyana, o modelo original ignorou o aspecto “duplo” da ZDP, em que jogadores mais experientes também deveriam ser afetados pelas interações com os menos experientes. A tentativa de aprimoramento de Levin e Cole (2007) consistiu na criação do procedimento *v-adjust-t*, em que o jogador mais experiente optaria por reduzir deliberadamente o seu desempenho na jogada seguinte, ajudando outros jogadores em dificuldades. Levin e Cole argumentaram que o novo procedimento modelaria a “consciência do professor quanto ao estudante” (ibid., p. 63).

A exposição do modelo P-V apontou questões essenciais das práticas científicas da modelagem em geral, e da vertente baseada em agentes: a necessidade de explicitação das suposições teóricas e intenções dos modeladores; o estabelecimento de campos comuns e da “língua franca” para o debate e colaboração entre pesquisadores; e as contribuições da modelagem para a investigação de





fenômenos psicológicos. A seguir, situaremos tais práticas no contexto amplo das possibilidades introduzidas pela Computação à atividade científica, como prelúdio das subseções que dissertarão sobre diferentes concepções, tipologias, processos de construção e avaliação de modelos.

Ciência computacional, modelagem e simulação

O aumento na disponibilidade de computadores para uso de pesquisadores a partir dos anos 1950 propiciou o desenvolvimento de métodos de análise do pensamento e do comportamento por meio de simulações computacionais. A premissa desses estudos era que o entendimento aprofundado de processos cognitivos seria necessário para que estes pudessem ser programados e executados em computadores, como no modelo P-V. Programas escritos para descrever processos cognitivos constituiriam teorias que, se fossem bem-sucedidas na comparação com o desempenho humano, seriam eficientes na predição de comportamentos em condições específicas e na variação daquelas condições (cf. HOVLAND, 1960; ABELSON, 1968; NEWELL; SIMON, 1972).

Desde meados dos anos 1970, modelagem e simulação computacionais são consideradas o “terceiro pilar” da atividade científica, ao lado de teoria e experimentação. A Computação tornou-se mais do que fornecedora de equipamentos, sendo entendida como um novo método para o pensamento e descoberta na ciência (WILSON, 1989; DENNING; MARTELL, 2015). Particularmente na Psicologia, Ostrom (1988) defendeu que pesquisadores habituados ao uso de linguagem natural e modelagem matemática em suas teorizações poderiam adotar a simulação computacional como o terceiro “sistema simbólico” da atividade científica.

Para Gilbert e Terna (2000), o aumento no uso de simulação computacional nas ciências sociais resultou de dois fatores: 1º) o desenvolvimento e popularização de ABM, tema deste capítulo; 2º) melhor apreciação e valorização de programas de computador como modelos de descoberta, compreensão e formalização. Sobre este aumento, Abrahamson e Wilensky (2005b) vislumbraram que a construção de simulações se tornaria prática comum nas investigações de fenômenos complexos pelas ciências naturais e sociais, e que as simulações seriam construídas, executadas e interpretadas pelos próprios pesquisadores dessas áreas, sem envolver programadores profissionais contratados para essas tarefas. Entretanto, relatos das discussões do modelo P-V por outros grupos de pesquisa sem conhecimentos de linguagens de programação sumarizaram dificuldades para realizar tal visão (ABRAHAMSON; WILENSKY; LEVIN, 2007; LEVIN; COLE, 2007).

De acordo Winsberg (2019), embora relacionadas a modelos, simulações computacionais teriam ao menos três concepções distintas. No sentido estrito, simulações computacionais são programas executados passo a passo por computadores para explorar comportamentos aproximados de modelos matemáticos. A especificação do estado inicial do sistema em um dado tempo t é informada como entrada para o programa, que calcula os estados subsequentes ($t+1$, $t+2$,...) produzindo uma representação numérica da evolução do estado do sistema. No sentido amplo, simulação computacional é um método completo para o estudo de sistemas, envolvendo a escolha, implementação e execução de modelos em computadores, seguida da análise, visualização e interpretação dos dados resultantes. A terceira concepção desvincula simulações de computadores, definindo-as como





qualquer sistema cujo comportamento dinâmico seja similar o bastante ao de outro sistema, de modo que o estudo do primeiro permitiria aprender algo sobre o segundo. A caracterização “computacional” implicaria a possibilidade de execução deste tipo de simulação por computadores.

Outra expressão, *simulação social*, é utilizada para denotar investigações que empregam métodos computacionais e rigor matemático para compreender sistemas sociais complexos (cf. GILBERT; TROITZSCH, 2005; COELHO, 2018). São simulações computacionais de fenômenos sociais, aplicando modelagem para investigar de que modo condutas individuais produzem resultados observados em eventos como manifestações, protestos e outras aglomerações.

Modelos e tipologias, estruturas e interpretações

Modelos consistem em estruturas abstratas ou físicas utilizadas para representar e analisar indiretamente fenômenos do mundo real (WEISBERG, 2007). Para Mooney e Swift (1999), modelos são representações *intencionais* da realidade, isto é, elaboradas com *propósitos* específicos. As intenções do modelador se relacionam, por um lado, com *restrições de recursos* – tempo, pessoas, orçamento e equipamentos disponíveis para a construção do modelo; e por outro com a *resolução* exigida para que o modelo possa ser útil na investigação do fenômeno de interesse.

Weisberg (2015) propôs três categorias de modelos: 1) *modelos concretos*, objetos físicos cujas propriedades potencialmente podem estar em relações de representação de fenômenos do mundo real – p.ex. maquetes de arquitetura, modelos de veículos em escala reduzida, inclusive brinquedos; 2) *modelos matemáticos*, estruturas abstratas

cujas propriedades potencialmente podem estar em relações com representações matemáticas de fenômenos – p.ex. equações, grafos; 3) *modelos computacionais*, conjuntos de procedimentos que potencialmente podem estar em relações com descrições computacionais do comportamento de sistemas – p.ex. algoritmos e programas de computador. O trecho “potencialmente podem estar em relações”, recorrente nas definições, alude à característica compartilhada pelas três tipologias. Na avaliação de Weisberg (2015, p. 15), com a qual concordamos e seguiremos nas alegações subsequentes, modelos são *estruturas interpretadas*, utilizadas por modeladores para *representar* fenômenos reais ou imaginados.

Começando pela *estrutura*, encontramos a noção de *alvo*, que são os sistemas para os quais o modelador pretende aplicar seus modelos. Estes são similares ao sistema alvo em aspectos importantes, embora permitam inferências mais gerais que o caso específico analisado. Há certa autonomia entre realidade e modelo, no juízo de que sua construção inicial independe de padrões existentes nos dados acerca do fenômeno, auxiliando a elaboração e exploração criativa de hipóteses. A estrutura mais relevante para modelos computacionais são algoritmos, conjuntos de instruções para executar procedimentos como *p-adjust*, *v-adjust*, *pv-adjust* e *r-adjust*.

Segundo Knuth (1997, p. 5), algoritmos são “conjuntos finitos de regras que fornecem sequências de operações para solucionar tipos específicos de problemas”. A finitude implica que algoritmos devem terminar após um número finito de passos, e as sequências de operações devem ser não ambíguas, precisamente definidas. Além disso, algoritmos têm zero ou mais *entradas*, que são valores fornecidos no início ou durante sua execução; e uma ou mais *saídas*, valores que





estão relacionados de modo específico às entradas – grosso modo, algoritmos *processam* entradas e *produzem* saídas. Por fim, espera-se que algoritmos sejam *eficazes*, formados por operações suficientemente básicas que possam ser realizadas com exatidão e em tempo finito por um indivíduo utilizando apenas papel e lápis.

Esta caracterização evidencia que algoritmos não são restritos à execução por computadores. Com efeito, estes dispositivos são programados para executar algoritmos como seres humanos, não obstante de forma mais rápida e com maior capacidade de processamento de entradas e saídas. O modelo P-V poderia ser simulado por pessoas lançando bolas de gude, desempenhando papéis nas quatro condições, ou pela manipulação de objetos físicos que representem a situação¹².

Estas breves ilustrações indicam que o mesmo modelo pode ser descrito por diversas estruturas, mas que cada uma teria capacidades distintas para representar o alvo. Isto nos leva à segunda parte da concepção modelos de Weisberg (2015) como estruturas *interpretadas*. A capacidade de o modelo M representar o alvo existe se e somente se o modelo (1) *satisfaz* (2) uma *função* que (3) *instancia* o alvo na (4) *interpretação do modelador*. As quatro partes deste argumento integram a explicação do potencial representacional do modelo (CUMMINS, 1989). Por *satisfação*, entende-se que M é capaz de assumir estados que podem representar valores interpretáveis como valores do alvo. Um *estado* de M consiste no vetor correspondendo a valores determinados para cada uma de suas variáveis. *Espaços de estados* referem-se ao conjunto de estados possíveis para o modelo, enquanto *trajetórias* dizem

12 O modelo de Schelling (1971) ilustra a execução de um algoritmo em turnos utilizando objetos físicos.

respeito às mudanças de estados no tempo, quando este é considerado. As *condições iniciais do modelo* determinam quais caminhos no espaço de trajetórias o modelo pode seguir, tendo em conta valores das variáveis independentes e conjuntos fixos de parâmetros. No caso de modelos sem evolução temporal, o modelador pode estar interessado no caráter estático ou estrutural do sistema – redes de relações, quantidade e densidade das conexões, entre outros.

De acordo com a tipologia do modelo, estados são formalizados por diferentes estratégias – pessoas desempenhando papéis; substituições de variáveis em equações; instruções em programas de computador. No modelo P-V, as sequências de instruções do algoritmo representam as regras do funcionamento cognitivo dos jogadores, enquanto as propriedades gráficas (p.ex. cor, forma, escala e posição) dos objetos representam jogadores, o objetivo (a linha alvo a ser atingida) e desfecho de cada lançamento.

A função f a ser satisfeita pelos estados de M é aquilo que o modelador *interpreta* como constitutivo do alvo, possivelmente ponderando sobre propósitos, restrições de recursos e resolução almejada. Ao sugerirmos que o aprendiz piagetiano do modelo P-V “aprende” com a sua tentativa anterior, e que o vygotskyano “aprende” com os vizinhos, estamos descrevendo duas funções distintas, doravante denominadas funções f_p e f_v , que deverão ser satisfeitas pelos estados do modelo. Os estados de M *instanciam* a função f do alvo se for possível estabelecer uma interpretação dos *mapeamentos* entre os elementos de M e do alvo.

Antes de desenvolver os conceitos de instanciação e mapeamento, faz-se necessário abordar a distinção entre o modelo e sua descrição. O modelador observa ou imagina o





espaço de estados (ou trajetória) do alvo e elabora descrições que *especificam* o modelo. Essas descrições podem ser verbais, algébricas ou procedimentos em linguagens de programação, e seu nível de precisão ou imprecisão pode especificar de um único modelo a famílias inteiras de modelos. Weisberg (2007) explicou que a relação entre modelos e suas descrições é do tipo muitos-muitos, no sentido de que haveria inúmeros modelos especificados por uma mesma descrição e muitas descrições *realizáveis* pelo mesmo modelo. Em outros termos, de acordo com a interpretação e intenções do modelador, os procedimentos do modelo P-V que realiza a descrição da condição piagetiana poderiam igualmente realizar descrições típicas de retroalimentações cibernéticas (cf. MILLER; GALANTER; PRIBRAM, 1960), e os procedimentos da condição vygotskyana poderiam descrever dimensões de fenômenos de influência e conformidade social (cf. MILGRAM, 1974).

De volta à questão da função f instanciada pelo modelo M , descrições instanciam modelos pela atribuição de valores aos seus parâmetros. Abrahamson e Wilensky (2005a) programaram, na linguagem NetLogo, conjuntos de procedimentos com potencial relação com descrições computacionais dos comportamentos de aprendizagem – as funções f_p e f_v já citadas. Cada conjunto de valores atribuídos aos parâmetros¹³ do modelo P-V *instanciam* f_p e f_v , permitindo ao modelador buscar mapeamentos satisfatórios entre a interpretação que fez das teorias, as condutas dos jogadores durante a brincadeira, e a interpretação da execução do modelo realizando aquelas descrições.

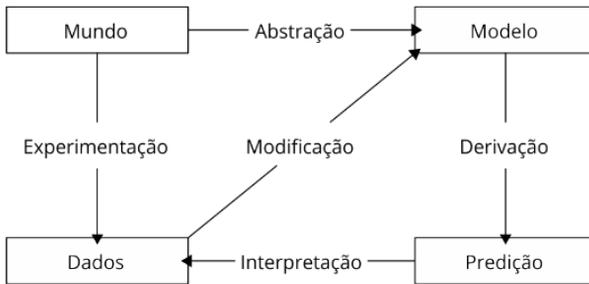
¹³ Parâmetros apresentados na descrição do modelo P-V: tentativas por turno, estratégia de aprendizagem, erro (ruído), número de jogadores, limiar da ZDP e número de vizinhos observados na ZDP.

O processo de modelagem baseada em agentes – ABM

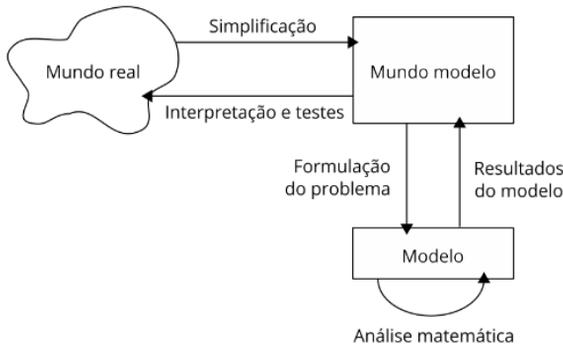
De acordo com Coombs, Dawes e Tversky (1970), o processo científico consiste na construção de modelos do mundo ou sistema alvo por abstrações, privilegiando certas propriedades do fenômeno de interesse e ignorando outras (Figura 4-a).

Figura 4 – Esquemas para os processos científico (a) e de modelagem (b).

a.



b.



Fonte: (a) adaptados de Coombs et al (1970, p. 3)
e (b) de Mooney e Swift (1999, p. 4).

A construção de modelos envolve isolar um conjunto de fenômenos empíricos, selecionar um sistema formal (p.ex. álgebra ou linguagens de programação) e estabelecer a correspondência





entre eles. As consequências do modelo construído podem ser derivadas empregando-se ferramentas da lógica e matemática, e as previsões obtidas são então comparadas e interpretadas quanto aos dados experimentais disponíveis, para determinar o grau de coordenação entre o modelo e o mundo. Esta etapa pode levar a modificações, revisões e extensões do modelo, de modo a incorporar novos dados.

Em perspectiva similar, Mooney e Swift (1999) apontaram que modeladores buscam resolver problemas no mundo real, marcado por complexidades e dificuldades. Por isso, o primeiro passo do processo de modelagem (Figura 4-b) corresponde à criação de um “mundo modelo”, que descarta boa parte da complexidade do mundo real. A tarefa de criação do mundo modelo é uma das mais desafiadoras do processo, pois requer decisões não triviais sobre o que precisa ser mantido ou descartado de modo a controlar a complexidade, sem comprometer o propósito da modelagem. Em seguida, cria-se um modelo do problema no mundo modelo, sobre o qual são aplicados métodos e técnicas da matemática para responder perguntas naquela simplificação da realidade. O passo final consiste em interpretar as respostas encontradas no mundo modelo, transportando-as de volta à realidade, evitando-se o equívoco de assumir que a resposta obtida no mundo modelado responde à questão posta no mundo real.

A modelagem baseada em agentes, ou *Agent-Based Modeling* (ABM), é uma abordagem de modelagem do tipo computacional, em que indivíduos ou “agentes” são descritos como entidades autônomas únicas que interagem entre si e com o ambiente de modo local (RAILSBACK; GRIMM, 2019). Agentes são únicos na medida em que têm propriedades distintas uns dos outros

(posição, tamanho, cor, histórico), e são autônomos porque atuam de forma independente, segundo seus objetivos e propósitos. A interação local significa que agentes restringem suas interações com a vizinhança a cada momento, não com todo o sistema. Em vez de manipular variáveis para resolver equações, como em modelos matemáticos, o modelador ABM define as características do ambiente e agentes, seus propósitos e regras de conduta, descritas como procedimentos de linguagens de programação. Depois, o modelador executa o modelo variando os parâmetros das condições iniciais, e observa a dinâmica emergente entre agentes e o meio no tempo.

ABM tem raízes nos estudos em teoria da complexidade e de sistemas adaptativos complexos originados na Biologia, nas investigações de autômatos celulares na Ciência da Computação, e nos avanços da Inteligência Artificial Distribuída (SMITH; CONREY, 2007). Destas, as duas primeiras são relevantes para a presente discussão. Dos estudos da complexidade, ABM herdou o interesse pela análise de sistemas *bottom-up*, onde regras simples produzem estruturas complexas e padrões emergentes. Nas obras introdutórias sobre ABM, figuram exemplos como as dinâmicas de cupins e cupinzeiros, formigas e formigueiros, pássaros e revoadas, veículos em engarrafamentos, predadores e presas, e indivíduos em sistemas de trocas econômicas simples (RESNICK, 1994; RAILSBACK; GRIMM, 2019). De autômatos celulares, ABM incorporou a concepção de agentes operando por regras simples em resposta às perturbações do meio. Nesses modelos, representa-se autômatos como pontos (*pixels*) sobre uma malha bidimensional, e altera-se suas cores para representar mudanças de estado – p.ex. preto





para “vivo”, branco para “morto” – em função da densidade da vizinhança¹⁴ (WOLFRAM, 2002).

Abrahamson e Wilensky (2005b) sustentaram que a prática de ABM envolve a exploração contínua de vastos intervalos de parâmetros das configurações ambientais e lógica dos comportamentos dos agentes, até que a simulação produza padrões semelhantes a observações empíricas do mundo real. Apesar de a relação de similaridade virtual-real não consistir em prova apropriada para as regras formuladas, permite investigar a plausibilidade delas e criar programas para tentar validá-las, seja pelo estudo sistemático do espaço de estados do modelo, seja por métodos complementares. Para estes autores, a estrutura metodológica de ABM teria atributos únicos: a) compartilharia e ampliaria o rigor dos experimentos de laboratório; b) o observador estaria “invisível” para os participantes do experimento de formas inacessíveis a intervenções etnográficas; c) permitiria conduzir com rapidez simulações de estudos longitudinais, ao mesmo tempo em que poderia analisar fenômenos breves em sequências de ações discretas; d) integraria as perspectivas *micro* e *macro*, articulando dados coletados de participantes únicos àqueles coletados com milhares de participantes; e) permitiria investigações do comportamento humano em condições que representariam riscos físicos e emocionais a participantes reais.

O processo de modelagem em geral, incluindo ABM, envolve a tradução de ideias intuitivas e observações empíricas, declaradas verbalmente, para descrições menos ambíguas, isto é, especificações baseadas em conceitos e ferramentas

14 O simulador online Francisco permite visualizar animações de autômatos celulares com regras de 8 bits. Disponível em <<https://www.hugocristo.com.br/projetos/francisco/>>. Acesso em: 05 mar. 2022.

matemáticas e princípios computacionais (VAN ROOIJ; BLOKPOEL, 2020). O ciclo de modelagem de Railsback e Grimm (2019) organiza tal tradução em seis etapas: 1) formulação da questão de pesquisa com precisão; 2) reunião de hipóteses sobre processos e estruturas que integrarão o modelo, descrevendo agentes, suas condutas e o ambiente em que atuarão; 3) escolha da estrutura do modelo – escala, entidades, variáveis de estado, processos e parâmetros – e de sua especificação verbal; 4) implementação do modelo, utilizando matemática e linguagens de programação para traduzir as descrições verbais no objeto “animado”; 5) análises, testes e revisões do modelo, explicitando o que pode ser aprendido com sua execução, explorando o espaço de parâmetros, e verificando que tipos de explicações ou predições são possíveis; 6) comunicação do modelo, seja via publicações científicas, seja disponibilizando o modelo para a comunidade.

A especificação verbal da terceira etapa do ciclo de modelagem pode ser feita com base no protocolo *Overview, Design concepts, and Details* – ODD (visão geral, conceitos de design e detalhes), elaborado por Grimm, Polhill e Touza (2017). Este protocolo padronizou descrições de modelos ABM para facilitar sua apresentação e entendimento, incluindo detalhamentos dos princípios teóricos subjacentes, medidas dos objetivos dos agentes, suas capacidades sensoriais e interações possíveis. Na época da publicação do modelo P-V, este papel foi cumprido pela aba Informações (Figura 1-3) de NetLogo.

A quinta etapa, que contempla análises, testes e revisões, adota técnicas variadas para avaliar a adequação do modelo e comparar suas saídas com dados empíricos relativos ao fenômeno no mundo real. Axtell e Epstein (1994) defenderam que modelos teriam critérios de sucesso distribuídos em quatro





níveis de desempenho e análise. No nível 0 (zero), o modelo é tratado como caricatura da realidade, proporcionando a visualização da aparência do fenômeno – p.ex. reproduzindo movimentos dos agentes no mundo real. No nível 1, o modelo produz concordância *qualitativa* com macroestruturas empíricas, estabelecidas pela inspeção visual de gráficos de distribuições de propriedades dos agentes simulados e reais. No nível 2, o modelo produz concordância *quantitativa* com macroestruturas empíricas, simulando o fenômeno real com precisão atestada por análises estatísticas. Por fim, no nível 3 obtém-se concordância *quantitativa* com estruturas empíricas macro e micro, de modo que o fenômeno real é simulado com precisão nas escalas agregada e individual, longitudinal e transversalmente. Os autores ainda relataram que a satisfação de um nível significa a satisfação dos critérios dos níveis subjacentes.

No entendimento de Weisberg (2007), a avaliação da adequação envolve intenções do modelador em quatro partes, nas quais a relação com dados empíricos integra mas não determina todos os juízos sobre o modelo: 1) *designação*, referente à especificação do fenômeno a ser estudado e sua coordenação matemática explícita entre partes do mundo real e partes do modelo, observando partes da estrutura do modelo inexistentes¹⁵ no alvo e que devem ser ignoradas; 2) *escopo*, relativo aos aspectos do fenômeno que seriam representados no modelo, implicando o alcance e limitações da construção pretendida; 3) *fidelidade dinâmica*, que informa o quão próximas as saídas do modelo e do fenômeno devem ser, incluindo margens de erro aceitáveis; 4) *fidelidade representacional*, que

15 A existência de elementos irrelevantes no modelo para a investigação do fenômeno no mundo real pode resultar da busca por fidelidade dinâmica e representacional entre eventos comuns aos dois domínios.

fornece padrões para avaliar se o modelo realiza previsões corretas sobre o sistema alvo pelos motivos corretos.

Adequação e avaliação do modelo

Os autores definiram o modelo P-V como um “experimento mental” para elucidar questões persistentes do debate entre adeptos das teorias de aprendizagem construtivista e socioconstrutivista. Considerando todas as simplificações que restringiram o escopo e fidelidade do modelo P-V, quando comparado aos contextos investigados por Piaget e Vygotsky, quais explicações do fenômeno da aprendizagem no mundo real puderam ser obtidas pela construção do modelo e execução das simulações?

Em primeiro lugar, a tradução dos pressupostos das teorias em linguagem natural para procedimentos computacionais exigiu a formalização precisa dos mecanismos de aprendizagem – os “ajustes”. Os resultados de cada tentativa são certamente distintos na simulação e no mundo real, mas os processos para obtê-los seriam *qualitativamente* similares. Ou seja, parâmetros e procedimentos que calculam a diferença entre a posição atingida e a linha alvo *instanciam* as funções cognitivas de assimilação, acomodação e da ZDP, com fidelidade suficiente para os propósitos do modelo P-V e de seus autores.

Em segundo lugar, na discussão dos resultados experimentais das simulações, Abrahamson e Wilensky (2005b) interpretaram os dados a partir dos pressupostos de cada teoria e respectivos achados empíricos. Os autores relataram que a condição aleatória (4) obteve o pior resultado, e que a “aprendizagem” ocorreu tanto nas condições piagetiana (1) quanto vygotskyana (2), sendo a condição combinada (3) a de melhor desempenho. Variações nos parâmetros erro, ZDP e





número de vizinhos definiram situações em que as condições 1 e 2 obtiveram desempenho superior. Outro aspecto relatado diz respeito aos efeitos conjuntos do tamanho da ZDP e do erro: valores maiores para a ZDP ajudaram mais jogadores a acertarem o alvo, pela imitação daqueles mais experientes; já valores menores para a ZDP e erros fizeram com que jogadores com desempenho inicial baixo não conseguissem melhorar – “ficaram para trás”. Ao final, os autores sugeriram que quanto maior o número de vizinhos observados, mais rápido ocorreria o aprendizado no grupo, pelo aumento nas chances de imitação de bons desempenhos.

Estas inferências ilustram a transposição do problema no mundo modelo para o real, o nível qualitativo de desempenho esperado, os critérios de adequação, as avaliações e testes do ciclo de modelagem (AXTELL; EPSTEIN, 1994; MOONEY; SWIFT, 1999; WEISBERG, 2007; RAILSBACK; GRIMM, 2019). As explicações acima decorrem das interpretações dos autores sobre os pressupostos das teorias, da exploração teoricamente orientada do espaço de parâmetros do modelo, e de mapeamentos entre padrões empíricos qualitativos observados e distribuições das propriedades dos agentes simulados.

Considerações finais

Modelos são estruturas interpretadas, representações intencionais e indiretas da realidade, ainda pouco usados em Psicologia por falta de formação específica. Desde os anos 1970, modelagem e simulação computacionais são consideradas o “terceiro pilar” da atividade científica, ao lado de teoria e experimentação. A premissa da modelagem é que a similaridade entre modelo e mundo real seja suficiente para que se possa

aprender algo sobre o segundo por meio do estudo do primeiro. A modelagem baseada em agentes (ABM) emprega regras simples para simular computacionalmente a conduta de agentes únicos e autônomos, que interagem entre si e com o ambiente localmente, produzindo efeitos emergentes. O processo de modelagem consiste no ciclo iterativo que traduz teorizações verbais em formalizações matemáticas e computacionais, culminando na implementação e disseminação do modelo para discussão pela comunidade. Há diferentes níveis e critérios de desempenho e de avaliação da adequação de modelos, relativos às intenções do modelador, recursos disponíveis e resolução para responder questões sobre o fenômeno de interesse. NetLogo é uma linguagem de programação e ambiente gratuito para a construção de modelos em ABM, oferecendo campo comum para o debate científico sobre teorias psicológicas. O modelo “*I’m Game!*”, baseado nas teorias de Piaget e Vygotsky e implementado em NetLogo, ilustra o potencial de ABM para a condução de estudos do desenvolvimento humano.

Referências

ABELSON, R. P. Simulation of Social Behavior. Em: **Handbook of Social Psychology**. 2. ed. Reading, Massachusetts: Addison-Wesley, 1968. v. 2p. 274–356.

ABRAHAMSON, D.; WILENSKY, U. **NetLogo Piaget-Vygotsky Game model**. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University, 2005a.

ABRAHAMSON, D.; WILENSKY, U. **Piaget? Vygotsky? I’m Game! – Agent-Based Modeling for Psychology Research**. . Em: THE 35TH ANNUAL MEETING OF THE JEAN PIAGET SOCIETY. Vancouver, Canada: jun. 2005b. Disponível em: <<https://gse.soe.berkeley.edu/faculty/dabrhamson>>. Acesso em: 26 jan. 2022





ABRAHAMSON, D.; WILENSKY, U.; LEVIN, J. A. **Agent-Based Modeling as a Bridge Between Cognitive and Social Perspectives on Learning**. Learning Complexity. *Anais...* Em: ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION. Chicago, IL: 2007.

AXTELL, R.; EPSTEIN, J. Agent-based modeling: Understanding our creations. *The Bulletin of the Santa Fe Institute*, v. 9, n. 4, p. 28–32, 1994.

CASTORINA, J. A. The Ontogenesis of Social Representations: A Dialectic Perspective. *Papers on Social Representations*, v. 1, n. 19, 17 mar. 2010.

COELHO, H. Simulação Social. *Revista de Ciência Elementar*, v. 6, n. 4, 2018.

COLE, M.; WERTSCH, J. V. Beyond the individual-social antinomy in discussions of Piaget and Vygotsky. *Human Development*, v. 39, n. 5, p. 250–256, 1996.

COOMBS, C. H.; DAWES, R. M.; TVERSKY, A. **Mathematical Psychology: An Elementary Introduction**. New Jersey: Prentice-Hall, 1970.

CUMMINS, R. **Meaning and Mental Representation**. Cambridge: MIT Press, 1989.

DE LA TAILLE, Y.; DE OLIVEIRA, M. K.; DANTAS, H. **Piaget, Vigotski, Wallon: teorias psicogenéticas em discussão**. São Paulo: Summus Editorial, 1992.

DENNING, P. J.; MARTELL, C. H. **Great Principles of Computing**. Cambridge: MIT Press, 2015.

GILBERT, N.; TERNA, P. How to build and use agent-based models in social science. *Mind & Society*, v. 1, n. 1, p. 57–72, 1 mar. 2000.

GILBERT, N.; TROITZSCH, K. **Simulation for the Social Scientist**. 2. ed. Maidenhead: Open University Press, 2005.

GRIMM, V.; POLHILL, G.; TOUZA, J. Documenting Social Simulation Models: The ODD Protocol as a Standard. Em: EDMONDS, B.; MEYER, R. (Eds.). **Simulating Social Complexity: A Handbook**.

Understanding Complex Systems. Cham: Springer International Publishing, 2017. p. 349–365.

GUEST, O.; MARTIN, A. E. How Computational Modeling Can Force Theory Building in Psychological Science. **Perspectives on Psychological Science**, p. 1745691620970585, 22 jan. 2021.

HOVLAND, C. I. Computer simulation of thinking. **American Psychologist**, v. 15, n. 11, p. 687–693, 1960.

KNUTH, D. **The Art of Computer Programming**. 3. ed. Massachusetts: Addison-Wesley, 1997. v. 1

LEVIN, J. A.; COLE, M. **Simulations as Mediators for Distributed Research Activity**. Learning Complexity. **Anais...** Em: ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION. Chicago, IL: 2007.

MILGRAM, S. **Obedience to Authority: an experimental view**. New York: Harper Perennial, 1974.

MILLER, G. A.; GALANTER, E.; PRIBRAM, K. H. **Plans and the Structure of Behavior**. New York: Holt, Rinehart and Winston, Inc., 1960.

MOONEY, D. D.; SWIFT, R. J. **A Course in Mathematical Modeling**. Washington, DC: The Mathematical Association of America, 1999. v. 13

NEWELL, A.; SIMON, H. A. **Human problem solving**. Englewood Cliffs, NJ: Prentice-Hall, 1972.

OSTROM, T. M. Computer simulation: The third symbol system. **Journal of Experimental Social Psychology**, v. 24, n. 5, p. 381–392, Setembro 1988.

PIAGET, J. O papel da equilibração na explicação psicológica. Em: **Seis estudos de Psicologia**. 24. ed. Rio de Janeiro: Forense Universitária, 2010a. p. 87–98.

PIAGET, J. Gênese e estrutura na psicologia da inteligência. Em: **Seis estudos de Psicologia**. 24. ed. Rio de Janeiro: Forense Universitária, 2010b. p. 121–132.





RAILSBACK, S. F.; GRIMM, V. **Agent-Based and Individual-Based Modeling: A practical introduction**. Princeton and Oxford: Princeton University Press, 2019.

RESNICK, M. **Turtles, Termites, and Traffic Jams: Explorations in massively parallel microworlds**. Cambridge: MIT Press, 1994.

SCHELLING, T. C. Dynamic models of segregation. **The Journal of Mathematical Sociology**, v. 1, n. 2, p. 143–186, 1 jul. 1971.

SMALDINO, P. E. How to Translate a Verbal Theory Into a Formal Model. **Social Psychology**, v. 51, n. 4, p. 207–218, 1 jul. 2020.

SMITH, E. R.; CONREY, F. R. Agent-Based Modeling: A New Approach for Theory Building in Social Psychology. **Personality and Social Psychology Review**, v. 11, n. 1, p. 87–104, Fevereiro 2007.

THAGARD, P. **Mente: introdução à ciência cognitiva**. Porto Alegre: Artmed, 1998.

TISUE, S.; WILENSKY, U. **Netlogo: A simple environment for modeling complexity**. . Em: INTERNATIONAL CONFERENCE ON COMPLEX SYSTEMS. Boston, MA, 2004.

VALLACHER, R. R.; NOWAK, A.; READ, S. J. Rethinking Human Experience: The Promise of Computational Social Psychology. Em: **Computational Social Psychology**. New York: Routledge, 2017.

VAN ROOIJ, I. Psychological models and their distractors. **Nature Reviews Psychology**, p. 1–2, 11 fev. 2022.

VAN ROOIJ, I.; BLOKPOEL, M. Formalizing verbal theories: A tutorial by dialogue. **Social Psychology**, v. 51, n. 5, p. 285–298, 2020.

VIGOTSKY, L. S. **A formação social da mente: o desenvolvimento dos processos psicológicos superiores**. 7. ed. São Paulo: Martins Fontes, 2007.

WEISBERG, M. Who is a Modeler? **The British Journal for the Philosophy of Science**, v. 58, n. 2, p. 207–233, 1 jun. 2007.

WEISBERG, M. **Simulation and Similarity: Using Models to Understand the World**. Oxford, New York: Oxford University Press, 2015.

WERTSCH, J. V. **Vygotsky and the Social Formation of Mind**. Cambridge: Harvard University Press, 1985.

WILENSKY, U. **NetLogo (and NetLogo User Manual)**. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, 1999.

WILSON, K. G. Grand challenges to computational science. **Future Generation Computer Systems**, Grand Challenges to Computational Science. v. 5, n. 2, p. 171–189, Setembro 1989.

WINSBERG, E. Computer Simulations in Science. Em: ZALTA, E. N. (Ed.). **The Stanford Encyclopedia of Philosophy**. Winter 2019 ed. Stanford, CA: Metaphysics Research Lab, Stanford University, 2019.

WOLFRAM, S. **A New Kind of Science**. Champaign: Wolfram Media, 2002.

