



UNIVERSIDADE ESTADUAL PAULISTA

“JÚLIO DE MESQUITA FILHO”

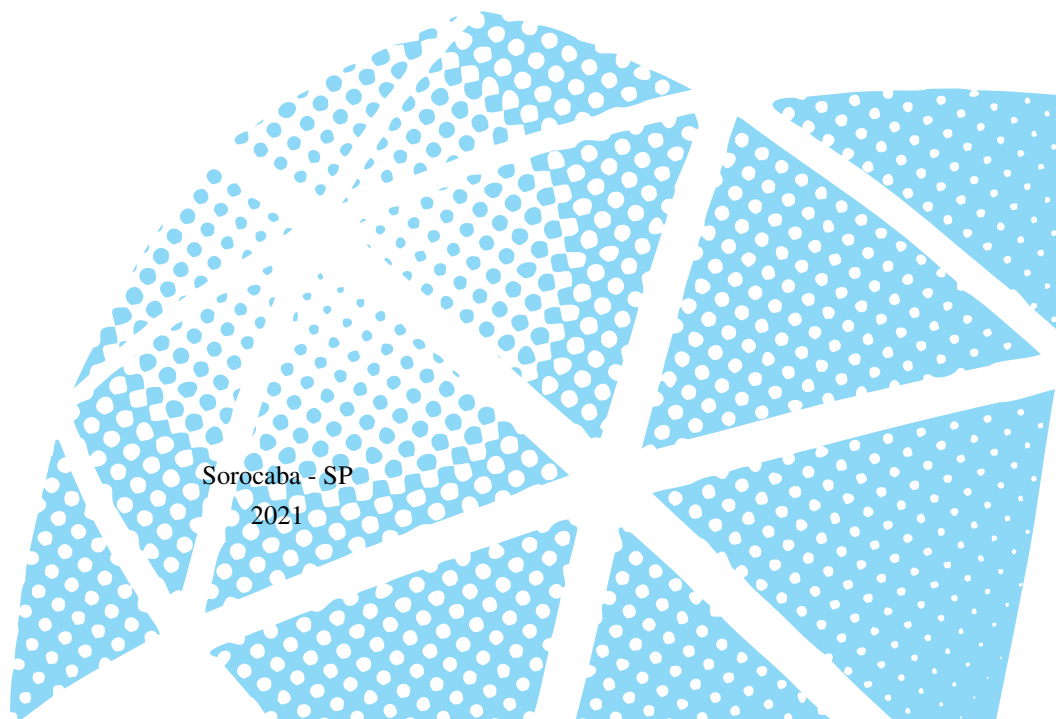
Instituto de Ciência e Tecnologia de Sorocaba

LEONARDO DE LELLIS ROSSI

**APRENDIZADO SENSORIO-MOTOR EM ROBÔS COGNITIVOS UTILIZANDO MODELO
CST-CONAIM**

Sorocaba - SP

2021

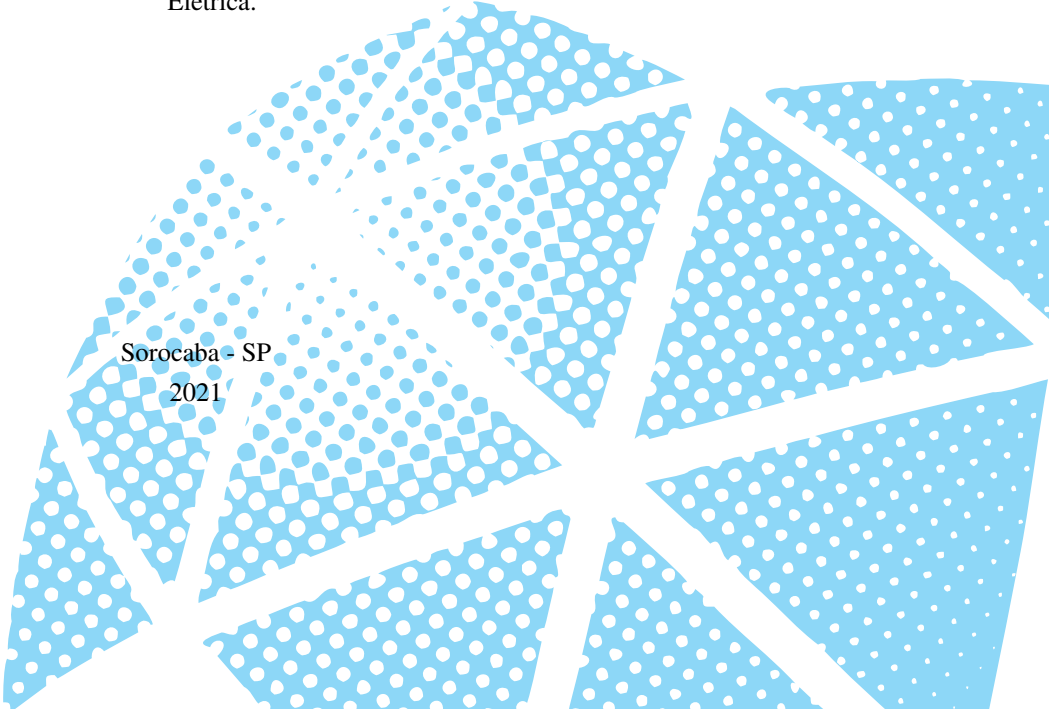


LEONARDO DE LELLIS ROSSI

Aprendizado sensório-motor em robôs cognitivos utilizando modelo CST-CONAIM

Texto apresentado ao Programa de Pós-Graduação em Engenharia Elétrica (PGEE) do Instituto de Ciência e Tecnologia de Sorocaba como parte dos requisitos para obtenção do título de Mestre em Engenharia Elétrica.

Sorocaba - SP
2021



R832a Rossi, Leonardo de Lellis
Aprendizado sensório-motor em robôs cognitivos utilizando modelo
cst-conaim / Leonardo de Lellis Rossi. -- Sorocaba, 2021
134 p. : il., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp),
Instituto de Ciência e Tecnologia, Sorocaba
Orientador: Alexandre da Silva Simões
Coorientador: Esther Luna Colombini

1. Engenharia Elétrica. 2. Automação. 3. Inteligência artificial. 4.
Cognição. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca do Instituto de
Ciência e Tecnologia, Sorocaba. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

CERTIFICADO DE APROVAÇÃO

TÍTULO DA DISSERTAÇÃO: Aprendizado sensório-motor em robôs cognitivos utilizando modelo CST-CONAIM

AUTOR: LEONARDO DE LELLIS ROSSI

ORIENTADOR: ALEXANDRE DA SILVA SIMÕES

COORIENTADOR: ESTHER LUNA COLOMBINI

Aprovado como parte das exigências para obtenção do Título de Mestre em ENGENHARIA ELÉTRICA, área: Automação pela Comissão Examinadora:

Prof. Dr. ALEXANDRE DA SILVA SIMÕES (Participação Virtual)
Departamento de Engenharia de Controle e Automação / Instituto de Ciência e Tecnologia - UNESP - Câmpus de Sorocaba

Prof. Dr. RICARDO RIBEIRO GUDWIN (Participação Virtual)
Departamento de Engenharia de Computação e Automação Industrial / Universidade Estadual de Campinas - Faculdade de Engenharia Elétrica e de Computação

Prof.^a Dr.^a DIANA FRANCISCA ADAMATTI (Participação Virtual)
Centro de Ciências Computacionais / Universidade Federal do Rio Grande - FURG

Sorocaba, 05 de julho de 2021

Prof. Dr. Alexandre da Silva Simões

A handwritten signature in black ink, consisting of a large, stylized loop followed by a long, sweeping horizontal stroke.

A todos que lutam pelo acesso à Educação, à Pesquisa, à Ciência e à Tecnologia no Brasil.

(In Memoriam)

Wilson Luiz de Lellis (1967-2020).

Oscar Camilo de Lellis (1935-2021).

Agradecimentos

Meus agradecimentos a todos os familiares, amigos, professores e funcionários do Instituto de Ciência e Tecnologia de Sorocaba e da Universidade de Campinas, que direta ou indiretamente contribuíram para a realização deste trabalho. A todos que acreditaram e contribuíram com sua palavra de apoio. Dedico meus agradecimentos:

- Aos meus pais Marta Valéria e Edson e à minha irmã Beatriz por todo o amor, apoio, incentivo e confiança. Por me acompanharem durante toda a minha vida;
- Aos meus familiares e amigos, pelo amor, apoio, incentivo e confiança. Por todo seu contato e convivência, seja pessoal ou virtual, por me ouvirem, me motivarem e me ampararem em todos os momentos. Por todo apoio e toda força, por toda a paciência e atenção, por todas as risadas e rolês, por todo o amparo e conselhos;
- À Letícia Berto, pela disponibilidade das tabelas de experimentos e por toda a ajuda com o desenvolvimento da arquitetura de integração entre o modelo CONAIM e as ferramentas do CST Toolkit;
- Aos meus amigos do Grupo de Automação e Sistemas Integráveis que me apoiaram e ajudaram na execução deste projeto;
- Aos meus maravilhosos orientadores, Prof. Dr. Alexandre da Silva Simões e Prof.^a Dr.^a Esther Luna Colombini, por todos os ensinamentos, incentivos e confiança. Por toda a sua atenção e paciência. Por todas as correções e conselhos, sejam eles técnicos ou pessoais;
- Ao Prof. Dr. Ricardo Gudwin, por todos os ensinamentos sobre arquiteturas cognitivas, pela colaboração com o CST Toolkit e por todas as discussões filosóficas no âmbito da robótica inteligente;
- Ao Prof. Dr. Eric Rohmer, por toda a ajuda com o simulador CoppeliaSim;
- À Prof.^a Dr.^a Paula Dornhofer Paro Costa, por todas as discussões a respeito de emoções artificiais;
- Aos meus colegas da RoboCup Brasil, em especial à Luciana Piccinini, à Patrícia Vergara, à Raíssa Cavalcanti, à Jussara Carvalho, à Prof.^a Dr.^a Cintia Kimie Aihara e à Prof.^a Dr.^a Sarah Thomaz, por todos os conselhos, conversas, motivações e apoio;
- Aos meus colegas da LenScope, em especial ao Marcos Makoto Ikegame e ao Luís Imagiire, por todos os conselhos, conversas, motivações e apoio;
- Ao grupo de teleacolhimento da Unesp, por todo o apoio durante a quarentena;
- À banca examinadora, Prof.^o Dr.^a Diana Francisca Adamatti e Prof. Dr. Ricardo Gudwin, por todos os comentários, sugestões e discussões quanto ao texto, de grande auxílio para a estruturação desta dissertação;
- À FAPESP pelo suporte financeiro do projeto PITEIBM-FAPESP, processo 16/18819-4, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) intitulado "Bootstrap Conditions for Interaction-based Multimodal Learning in Cognitive Robotics".

"Se você quer ser criativo, permaneça em parte como as crianças, com a criatividade e a imaginação que as caracterizam antes de serem deformadas pela sociedade adulta."

Jean Piaget

Resumo

Arquiteturas cognitivas computacionais, como o CONAIM (*Conscious Attention-Based Integrated Model* (SIMões, 2015; SIMões; COLOMBINI; RIBEIRO, 2016; SIMOES; COLOMBINI; RIBEIRO, 2017; COLOMBINI; SIMões; RIBEIRO, 2017)), sugerem que um caminho para a obtenção da consciência em máquinas seria o uso coordenado de diversos elementos inspirados naqueles presentes no sistema cognitivo humano, como memórias, raciocínio, planejamento, tomada de decisão, aprendizado, motivação e atenção. Contudo, a implementação desse tipo de sistema tem se mostrado complexa, demandando ferramentas computacionais com alto grau de especialização ou aplicáveis em situações muito limitadas. Uma plataforma especializada para a construção de arquiteturas cognitivas, o CST (*Cognitive Systems Toolkit* (GUDWIN et al., 2013; PARAENSE et al., 2016)), tem se destacado nesse cenário. Nela estão disponíveis estruturas cognitivas elementares denominadas *codelets* e objetos de memória (*memory objects*), capazes de dar suporte à implementação dos módulos previstos no modelo CONAIM, bem como suas possíveis evoluções. Trabalhos realizados pelo grupo de pesquisa implementaram módulos atencionais do CONAIM e módulos de aprendizado por reforço (AR) com as ferramentas disponíveis no CST (BERTO et al., 2020b; BERTO, 2020; REGATTIERI; COLOMBINI, 2018; ARAÚJO; GUDWIN, 2018) e delinearam uma série de experimentos cognitivos que podem contribuir para a investigação desse novo modelo e de seus impactos computacionais (BERTO, 2020). O presente trabalho busca estender essa investigação através da implementação de um sistema cognitivo-atencional com um único mecanismo de aprendizado procedimental, baseado no processo incremental de aprendizado humano nos três primeiros subestágios sensório-motores da Teoria de Jean Piaget (PIAGET, 1952). A validação do processo de desenvolvimento e aprendizado deste sistema foi efetuada através da realização de experimentos cognitivos propostos por Berto (2020), utilizando robôs móveis em ambiente simulado. Os resultados obtidos evidenciam a possibilidade de se avaliar o nível de aprendizado de um robô através de experimentos cognitivos incrementais.

Palavras-chave: Inteligência Artificial. Arquiteturas Cognitivas. Robótica do Desenvolvimento.

Abstract

Cognitive architectures, as the CONAIM model (*Conscious Attention-Based Integrated Model*) (SIMÕES, 2015; SIMÕES; COLOMBINI; RIBEIRO, 2016; SIMÕES; COLOMBINI; RIBEIRO, 2017; COLOMBINI; SIMÕES; RIBEIRO, 2017), suggest that a way to obtain consciousness in machines is the use of several factors inspired by those present in the human cognitive system, such as memories, reasoning, planning, decision making, learning, motivation, and attention. However, implementing this type of system has shown to be complex, requiring computational tools with a high degree of specialization or implemented in very limited situations. A specialized platform for building cognitive architectures, CST (*Cognitive Systems Toolkit*), stands out in this scenario (GUDWIN et al., 2013; PARANSE et al., 2016). It contains cognitive structures called *codelets* and *memory objects*, capable of supporting the modules provided in the CONAIM model, as well as their possible evolutions. Recent works of our research group implemented CONAIM and reinforcement learning (RL) modules with the CST (REGATTIERI; COLOMBINI, 2018; BERTO, 2020; BERTO et al., 2020a; BERTO et al., 2020b) with a single procedural learning mechanism that, without external modifications, demonstrates the incremental learning process in the first three sensorimotor substages in Jean Piaget's Theory (PIAGET, 1952). The validation of the development and learning process of this system will be carried out through a collection of cognitive experiments that can be used to study the new model and its computational and cognitive impacts (BERTO, 2020). Thus, the present work seeks to extend this investigation by implementing the learner attentive-cognitive agent and its validation through some of the proposed cognitive experiments using mobile robots in simulated environments. The results obtained show the possibility of evaluating the learning level of a robot through incremental cognitive experiments.

Keywords: Artificial Intelligence. Cognitive Architectures. Developmental Robotics.

Lista de Figuras

Figura 1 – a) Ilustração do processo de percepção, representação e ação em um sistema cognitivista . b) Ilustração do processo de percepção, representação e ação em um sistema conexionista . Fonte: (GUDWIN, 2014) (adaptado).	27
Figura 2 – Cronologia de 84 arquiteturas e estruturas cognitivas baseadas em representação da mente e processamento cognitivo.	28
Figura 3 – Modelo atencional de Colombini (2014) Fonte: (COLOMBINI, 2014) (adaptado).	35
Figura 4 – Contribuição no aumento e decaimento de estímulos atencionais ao longo do tempo. Fonte: (COLOMBINI, 2014).	37
Figura 5 – Sistema cognitivo-atencional CONAIM. Fonte: Simões (2015).	38
Figura 6 – Diagrama de classes do CST Toolkit. Fonte: (GUDWIN et al., 2013).	41
Figura 7 – <i>Memory Container</i> no CST Toolkit. Fonte: (BERTO, 2020) (adaptado).	42
Figura 8 – Núcleo do CST Toolkit. Fonte: (PARAENSE et al., 2016) (adaptado).	43
Figura 9 – Esquema ilustrando o processo de discretização do Mapa de Saliências. Fonte: (BERTO, 2020) (adaptado).	46
Figura 10 – Esquema de aprendizado procedimental com CONAIM <i>bottom-up attentional</i> implementado com CST. Fonte: (BERTO, 2020) (adaptado).	46
Figura 11 – a) Humanoide Marta no Laboratório de Automação e Sistemas Integráveis (LASI) - UNESP Sorocaba/SP. b) Humanoide Marta no Simulador CoppeliaSim. c) 25 Graus de Liberdade da humanoide. Fonte: (TOMAZELA; COLOMBINI, 2019). d) Robô Pioneer 3DX no Simulador CoppeliaSim.	60
Figura 12 – Posição do Sensor de Visão da Humanoide Marta no Simulador CoppeliaSim. Y: 1.99m e rotação em X: -20°.	61
Figura 13 – a) Imagem recebida através da RemoteAPI. Os eixos representam as coordenadas x e y de cada pixel. b) 256 regiões utilizadas no cálculo dos mapas de características para este trabalho. (c) Ilustração da conversão 3D para 2D.	63
Figura 14 – Simplificação da extração de características	65
Figura 15 – 16 regiões utilizadas como estados para o Aprendizado Procedimental neste trabalho.	67
Figura 16 – Esquema de implementação da arquitetura CONAIM+CST com a humanoide Marta e que recebe estímulos atencionais <i>bottom-up</i> e <i>top-down</i> .	68
Figura 17 – Diagrama de Classes para validação da arquitetura CONAIM+CST implementada com a humanoide Marta e que recebe estímulos atencionais <i>bottom-up</i> e <i>top-down</i> .	70
Figura 18 – (a) humanoide Marta (algoritmo cognitivo-atencional), Robô Pioneer 3DX (algoritmo de Braitenberg) e objetos estáticos no Simulador CoppeliaSim. (b) Motores utilizados (em vermelho) para os experimentos relacionados com a habilidade de aprender a rastrear e acompanhar objetos e outros agentes. Fonte: (TOMAZELA; COLOMBINI, 2019) (adaptado). (c) Posições para as quais a fôvea da agente pode ser posicionada.	72
Figura 19 – Modelo interno da agente humanoide Marta. O modelo é iniciado no EXP 1 somente com os elementos em azul. No EXP 2 são adicionados os elementos em vermelho . No EXP 3 são adicionados os elementos em amarelo .	73
Figura 20 – (1º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental.	84

Figura 21 – (1º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.	86
Figura 22 – (1º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios do Aprendizado Procedimental.	87
Figura 23 – (1º Subestágio) Dados sensoriais obtidos durante o Aprendizado Procedimental.	88
Figura 24 – (Berto 1º Subestágio - Exp. A) Dados sensoriais obtidos nas execuções do Exp. A. Objeto em posição fixa e de cor primária.	90
Figura 25 – (Berto 1º Subestágio - Exp. A) Médias e desvios padrão do número de ações executadas a cada 10 episódios.	91
Figura 26 – (Berto - 1º Subestágio - Exp. A) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. (Berto 1º Subestágio - Exp. A) Dados sensoriais obtidos nas execuções do Exp. A. Objeto em posição fixa e de cor primária.	92
Figura 27 – (Berto 1º Subestágio - Exp. B) Dados sensoriais obtidos nas execuções do Exp. B. Objeto se movimentando lentamente e de cor primária.	93
Figura 28 – (Berto 1º Subestágio - Exp. B) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando lentamente e de cor primária.	94
Figura 29 – (Berto - 1º Subestágio - Exp. B) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. Objeto se movimentando lentamente e de cor primária.	95
Figura 30 – (2º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental.	97
Figura 31 – (2º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número total de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.	98
Figura 32 – (2º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios do Aprendizado Procedimental.	99
Figura 33 – (2º Subestágio) Dados sensoriais obtidos no episódio 20 do Aprendizado Procedimental.	99
Figura 34 – (Berto - 2º Subestágio) Dados sensoriais obtidos nas execuções deste experimento. Objeto se movimentando e de cor primária.	100
Figura 35 – (Berto - 2º Subestágio) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando e de cor primária. Objeto se movimentando e de cor primária.	101
Figura 36 – (Berto - 2º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. Objeto se movimentando e de cor primária.	102
Figura 37 – (3º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental	105
Figura 38 – (3º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental.	106
Figura 39 – (3º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número total de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.	106
Figura 40 – (3º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios no Aprendizado Procedimental.	107
Figura 41 – (3º Subestágio) Dados sensoriais obtidos no episódio 180 do Aprendizado Procedimental.	108
Figura 42 – (Berto - 3º Subestágio) Dados sensoriais obtidos na primeira execução do experimento. Objeto se movimentando e de cor primária	109
Figura 43 – (Berto - 3º Subestágio) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando e de cor primária.	110

Figura 44 – (Berto - 3º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. Objeto se movimentando e de cor primária.	111
Figura 45 – Mudanças de ações executadas em um episódio teste em cada subestágio.	115

Lista de Tabelas

Tabela 1 – Comparação entre arquiteturas cognitivas. Adaptado de (SAMSONOVICH, 2010a).	48
Tabela 2 – Características do Estágio Sensório-Motor de Piaget	52
Tabela 3 – Experimentos relacionados com a habilidade de rastrear objetos. Fonte: (BERTO, 2020).	129

Lista de Abreviações e Siglas

Modelos, Arquiteturas e Ferramentas Cognitivas

ACT-R	<i>Adaptive Control of Thought - Rational</i>
ADAPT	<i>Adaptive Dynamics and Active Perception for Thought</i>
BBD	<i>Brain-Based Device</i>
CLARION	<i>Connectionist Learning with Adaptive Rule Induction On-line</i>
CONAIM	<i>Conscious Attention-Based Integrated Model</i>
CST	<i>Cognitive System Toolkit</i>
DRAMA	<i>Dynamic Recurrent Associative Memory Architecture</i>
EPIC	<i>Explicitly Parallel Instruction Computing</i>
FORR	<i>FOr the Right Reasons</i>
LIDA	<i>Learning Intelligent Distribution Agent</i>
MDB	<i>Multilevel Darwinist Brain</i>
SASE	<i>Self-Affecting Self-Aware</i>
Soar	<i>State, Operator, and Result</i>
SPA	<i>Semantic Pointer Architecture</i>

Elementos do CST Toolkit

MO	Objetos de Memória (<i>Memory Object</i>)
----	---

Elementos da arquitetura SOAR

EMT	Elementos da Memória de Trabalho
SVE	Sistema de Visão Espacial

Elementos da arquitetura CLARION

SCA	Subsistema centrado em ação
SNCA	Subsistema não-centrado em ação
SML	Subsistema motivacional
SMC	Subsistema metacognitivo

Elementos da arquitetura LIDA

MET	Memória Episódica Transitória
MPA	Memória Perceptual Associativa
MAS	Modelo Atual Situacional

Elementos e mecanismos computacionais

RL	Aprendizado por Reforço (<i>Reinforcement Learning</i>)
MDP	Markov Decision Process
IOR	Mecanismo atencional de inibição de retorno de localização ou objeto
WTA	Mecanismo de tomada de decisão <i>Winner-takes-all</i>

Elementos do ConsScale

CS	Habilidades Cognitivas (<i>Cognitive Skills</i>)
CLS	Nível de Pontuação Cumulativa (<i>Cumulative Level Score</i>)
CQS	Pontuação Quantitativa (<i>ConsScale Quantitative Score</i>)
Li	Observação de cada nível

Outras siglas

devRobotics	Developmental robotics ou Robótica epigenética
Bootstrap	Projeto PITE IBM-FAPESP, processo 16/18819-4, Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), "Bootstrap Conditions for Interaction-based Multimodal Learning in Cognitive Robotics"
ETG	Espaço de Trabalho Global
IIT	<i>Global Workspace Theory</i> (Teoria de Integração da Informação)
GWT	<i>Global Workspace Theory</i> (Teoria do Espaço de Trabalho Global)
HOT	<i>Higher-order Thought</i> (Teoria do Pensamento de Ordem Superior)
ZDP	Zona de Desenvolvimento Proximal

Lista de Símbolos

Elementos da arquitetura CONAIM

a	Ação do agente
ξ	Componente do Index da Característica Vencedora do elemento b_i no sistema atencional
$\mathcal{A}$	Conjunto de ações do agente
$\mathcal{B}$	Conjunto de Características Vencedoras no sistema atencional
$\mathcal{S}d$	Conjunto de dimensões de sensores
S	Conjunto de estados do agente
$\mathcal{F}$	Conjunto de Mapas de Características no sistema atencional
$\mathcal{G}$	Conjunto de objetivos
$\mathcal{O}$	Conjunto de observações das dimensões dos sensores no sistema atencional
$\mathcal{S}$	Conjunto de sensores
τ_{aa}	Constante de ativação ascendente exponencial no sistema atencional
τ_{ad}	Constante de ativação de decaimento exponencial no sistema atencional
τ_{eaa}	Constante de ativação exponencial endógena ascendente no sistema atencional
τ_{ead}	Constante de ativação exponencial endógena decrescente no sistema atencional
τ_{ra}	Constante de refração ascendente exponencial no sistema atencional
τ_{rd}	Constante de refração de decaimento exponencial no sistema atencional
τ_{ξ}	Constante relacionada à vizinhança da Característica Vencedora no sistema atencional
δ_p	Delay de pré-ativação constante no sistema atencional
δ_r	Delay de pré-refração constante no sistema atencional
δ_{edp}	Delay no decaimento endógeno no sistema atencional
sd_n	Elemento de dimensão do Conjunto de Dimensões de Sensores
b_i	Elemento do Conjunto de Características Vencedoras no tempo t com índice i , indicando a natureza do estímulo (endógeno ou exógeno), no sistema atencional
f_{2d_n}	Elemento do Conjunto de Mapas de Características no sistema atencional
g_t	Elemento do Conjunto de Objetivos
e_n	Elemento do Conjunto de Estados do Agente
s_n	Elemento do Conjunto de Sensores
m_n	Elemento do Mapa Atencional no sistema atencional
c_n	Elemento do Mapa de Características Combinado no sistema atencional
l_n	Elemento do Mapa de Saliências no sistema atencional
$\mathcal{W}_f$	Vetor de pesos de características no sistema atencional
wf_{2n}	Elemento do Vetor de Pesos de Características no sistema atencional
δ_{eep}	Excitação endógena pré-ativação no sistema atencional
α	Função de Tomada de Decisão no sistema atencional
H	Função degrau (função de Heaviside)
ϕ_{BU_n}	Funções de mapeamentos de características <i>bottom-up</i> no sistema atencional
ϕ_{TD_n}	Funções de mapeamentos de características <i>top-down</i> no sistema atencional
t	Instante da observação e tempo de ativação
$\mathcal{M}$	Mapa Atencional no sistema atencional .
$\mathcal{C}$	Mapa de características combinado no sistema atencional
$\mathcal{L}$	Mapa de Saliências no sistema atencional

Elementos da arquitetura CONAIM (cont.)

$\mathcal{D}$	Conjunto de regiões utilizadas no Aprendizado Procedimental
d	Elemento do conjunto de regiões utilizadas no Aprendizado Procedimental
r	Regiões utilizadas para a construção dos mapas de características <i>bottom-up</i> e <i>top-down</i>
$\mathcal{O}_{i,j,k}$	Observação da dimensão $j - t$ do sensor $i - t$ no tempo k
p	Pixels das imagens adquiridas através do sensor de visão da Humanoide Marta
$\mathcal{N}$	Conjunto de multiplicadores utilizados na discretização no Aprendizado Procedimental
d	Elemento do conjunto de multiplicadores utilizados na discretização no Aprendizado Procedimental

Elementos do CST Toolkit

E	Avaliação de um objeto de memória
Gi	Entrada Global de um codelet
Li	Entrada Local de um codelet
I	Informação de um objeto de memória
A	Nível de Ativação de um codelet
T	Registro de tempo de um objeto de memória
O	Saída Padrão de um codelet

Elementos do algoritmo de aprendizado *QLearning*

$Q_{old}(e_t, a_t)$	Atualização para a qualidade / utilidade da ação a_t no estado e_t
A	Conjunto de ações a_t
E	Conjunto de estados e_t
γ	Fator de desconto que determina a importância de recompensas futuras
$Q_{new}(e_t, a_t)$	Qualidade / utilidade da ação a_t no estado e_t
$\alpha(e_t, a_t)$	Taxa de aprendizado, que determina em que medida informações recém-adquiridas substituem informações antiga
r_t	Valor da recompensa para a ação a_t no estado e_t
$max_a Q(e_{t+1}, a)$	Valor máximo de recompensa em um novo estado e_{t+1} , considerando todas as ações A

Sumário

1	<i>INTRODUÇÃO</i>	21
1.1	Objetivos	22
1.1.1	Objetivo geral	22
1.1.2	Objetivos específicos	22
1.2	Contribuições	23
1.3	Organização do Texto	23
2	<i>ARQUITETURAS E ESTRUTURAS COGNITIVAS</i>	25
2.1	Revolução Cognitiva	25
2.2	Arquiteturas Cognitivas	26
2.3	Histórico e Taxonomia	26
2.4	Características	28
2.4.1	Percepção e Sensoriamento	28
2.4.2	Memórias	28
2.4.3	Atenção	30
2.4.4	Tomada de decisão	30
2.4.5	Aprendizado	31
2.5	CONAIM	32
2.5.1	Sistema atencional	32
2.5.2	Sistema cognitivo	37
2.5.3	Formalização	37
2.5.4	Operação	40
2.6	CST	41
2.6.1	<i>Codelets</i>	41
2.6.2	Objetos de memória	42
2.6.3	<i>Mind</i>	42
2.6.4	<i>Coderack</i>	43
2.6.5	<i>Raw Memory</i>	43
2.7	CONAIM+CST <i>bottom-up attentional</i>	44
2.7.1	Percepção e Atenção	44
2.7.2	Aprendizado Procedimental	45
2.8	Comparação entre Arquiteturas	47

3	ROBÓTICA DO DESENVOLVIMENTO	49
3.1	A Teoria de Piaget	49
3.2	Características da Robótica do Desenvolvimento	51
3.3	Princípios do desenvolvimento	53
3.4	Projetos relacionados	54
3.4.1	Interação social	54
3.4.2	Interação não-social	55
3.5	<i>Cognits</i>	57
3.6	A analogia de Berto	58
3.6.1	Experimentos Cognitivos	58
4	METODOLOGIA	59
4.1	Descrição geral	59
4.2	Materiais	59
4.2.1	Ambiente computacional	59
4.2.2	Agentes Robóticos	60
4.3	Modelagem da agente cognitivo-atencional segundo o CONAIM	61
4.3.1	Espaço Sensorial	61
4.3.2	Extração de Características <i>bottom-up</i>	62
4.3.3	Extração de Características <i>top-down</i>	64
4.3.4	Mapa de Características Combinado	65
4.4	Um modelo de aprendizado procedimental no sistema cognitivo-atencional CONAIM+CST	67
4.5	Modelo Atencional Top-Down+Bottom-up	68
4.6	Experimentos propostos	71
4.6.1	(EXP-01) 1º Subestágio: Uso de Reflexos	74
4.6.2	(EXP-02) 2º Subestágio: Reações Circulares Primárias	75
4.6.3	(EXP-03) 3º Subestágio: Reações Circulares Secundárias	79
5	RESULTADOS E DISCUSSÕES	83
5.1	(EXP-01) 1º Subestágio: Uso de Reflexos	83
5.1.1	Validação do Sistema Atencional	83
5.1.2	Validação do Sistema de Aprendizado Procedimental	85
5.1.3	Validação do Aprendizado com os Experimentos de Berto	89
5.1.3.1	<i>Experimento A - 1º Subestágio - Objeto em posição fixa e de cor primária</i>	89
5.1.3.2	<i>Experimento B - 1º Subestágio - Objeto se movimentando lentamente e de cor primária</i>	93
5.2	(EXP-02) 2º Subestágio: Reações Circulares Primárias	96
5.2.1	Validação do Sistema Atencional	96
5.2.2	Validação do Sistema de Aprendizado Procedimental	97
5.2.3	Validação do Aprendizado com os Experimentos de Berto - Objeto se movimentando e de cor primária	100
5.3	(EXP-03) 3º Subestágio: Reações Circulares Secundárias	103

5.3.1	Validação do Sistema Atencional	103
5.3.2	Validação do Sistema de Aprendizado Procedimental	104
5.3.3	Validação do Aprendizado com os Experimentos de Berto - Objeto se movimentando e de cor primária	109
5.4	Discussões	112
5.4.1	1º Subestágio	112
5.4.2	2º Subestágio	113
5.4.3	3º Subestágio	113
6	CONCLUSÃO	117
6.1	Considerações Finais	117
6.2	Objetivos alcançados	117
6.3	Publicações	118
6.4	Trabalhos Futuros	118
	REFERÊNCIAS	119
ANEXO A	<i>EXPERIMENTOS SENSORIO-MOTORES</i>	129
ANEXO B	<i>VALORES ABSOLUTOS DOS GRÁFICOS DE NÚMERO DE AÇÕES EXECUTADAS PARA CADA POSSÍVEL AÇÃO</i>	131

1 INTRODUÇÃO

"Ao invés de tentar produzir um programa para simular a mente adulta, por que não tentar produzir um programa que simule a mente de uma criança? Se essa mente fosse submetida a uma educação apropriada, seria possível obter uma mente adulta."

(Allan Turing)

(Computing Machinery and Intelligence.)

O desenvolvimento na espécie humana está fortemente relacionado à nossa capacidade de coordenar adequadamente um complexo conjunto de mecanismos mentais durante toda a nossa vida (PAPALIA; FELDMAN, 2013). O acesso às nossas memórias nos permite desenvolver conceitos e interpretações sobre o mundo. Através do estímulo do raciocínio e planejamento, organizamos processos e tomamos decisões. Por meio da comunicação, podemos aprender, ensinar, produzir e reproduzir conhecimento. Entretanto, antes que estas habilidades complexas pudessem surgir, algumas condições anteriores tiveram de ser alcançadas. Antes de começarmos a andar, tivemos que primeiro explorar e conhecer nosso próprio corpo, entender nossa capacidade motora e nossa interação com o ambiente. Conforme proposto por Piaget (PIAGET, 1952), o processo de desenvolvimento das habilidades cognitivas ocorre incrementalmente em nossas vidas.

A necessidade de compreender como a espécie humana incorpora novos conhecimentos e habilidades no decorrer de nossa existência é uma das motivações nas pesquisas que envolvem **teorias e modelos para a cognição** (CASTRO; LANDEIRA-FERNANDEZ, 2011; REGGIA, 2013; SIMÕES, 2015). O desenvolvimento nos campos da filosofia (CASTRO; LANDEIRA-FERNANDEZ, 2011; GENNARO, 2020), psicologia (WUNDT, 1897; PIAGET, 1952; JAMES, 1890; PAPALIA; FELDMAN, 2013), neurociência (KOCH; GREENFIELD, 2007; TONONI, 2016; BERLUCCHI; MARZI, 2019) e, mais recentemente, da ciência da computação e engenharia (REGGIA, 2013; SIMÕES, 2015) propuseram ideias, visões e ferramentas que contribuiriam para o estudo de modelos cognitivos aplicáveis em máquinas (BROOKS, 1991; REGGIA, 2013; SIMÕES, 2015).

O aumento do interesse de áreas como a inteligência artificial e a robótica por modelos dessa natureza estimulou a recente nucleação de uma área do conhecimento denominada **robótica do desenvolvimento** (*developmental robotics*) (LUNGARELLA; SANDINI; PFEIFER, 2003). A área, usualmente abreviada como **devRobotics**, ou ainda **robótica epigenética** (*epigenetic robotics*), tem buscado inspiração direta na interação entre corpo, mente e ambiente e no desenvolvimento do aprendizado e de habilidades sensorio-motoras em crianças (CANGELOSI; SCHLESINGER, 2014). As técnicas estudadas nesse contexto objetivam que robôs autônomos possam operar em cenários complexos, parcialmente desconhecidos, nos quais não é possível a pré-programação, e que exigem que os robôs tenham uma capacidade mais comparável à dos humanos para analisar e executar tarefas (LUNGARELLA; SANDINI; PFEIFER, 2003).

Pesquisadores da área da robótica do desenvolvimento têm se preocupado com a proposição de modelos computacionais que possam fornecer estruturas teóricas adequadas para trabalhar com processos cognitivos, inspiradas pela maneira como os humanos adquirem o conhecimento (SUN, 2004; FRANKLIN, 2012). Esses **modelos e arquiteturas cognitivas** requerem a coordenação de um conjunto de mecanismos que dependem uns dos outros e objetivam reproduzir comportamentos complexos da mente humana em entidades artificiais (THOMSON; BENNATI; LEBIERE, 2014; LAIRD, 2008). Essas arquiteturas e processos também são objeto de estudo de outro campo bastante próximo, o da **consciência de máquina** (*machine consciousness*), que busca trazer para o universo da automação processos que, por exemplo, permitem aos seres humanos se tornarem conscientes de sua existência e do mundo que os cerca (REGGIA, 2013; SIMÕES, 2015).

Dentre as maiores dificuldades encontradas na atualidade para o trabalho nestes campos do conhecimento estão: *i)* a maioria dos modelos e arquiteturas propostos para trabalhar com processos cognitivos possuem baixo grau de formalismo, o que pode gerar múltiplas possíveis interpretações e dificultar sua evolução incremental; *ii)* esses modelos são muitas vezes utilizados em tarefas muito específicas ou em ambientes excessivamente limitados para analisar o desenvolvimento cognitivo, não sendo adequados para abordagens ou testes computacionais complexos; *iii)* parte desses trabalhos tem sido desenvolvidos em arquiteturas fechadas; *iv)* a maioria dos modelos e arquiteturas não possuem processos claros de experimentação e avaliação.

Projetos do presente grupo de trabalho propuseram alguns dos primeiros modelos formais para lidar com *arquiteturas cognitivas e conscientes* (COLOMBINI, 2014; SIMÕES, 2015; SIMÕES; COLOMBINI; RIBEIRO, 2016; PARAENSE et al., 2016; SIMÕES; COLOMBINI; RIBEIRO, 2016; COLOMBINI; SIMÕES; RIBEIRO, 2017; SIMOES; COLOMBINI; RIBEIRO, 2017), *kits de ferramentas e softwares cognitivos* (CAPITANIO; SILVA; GUDWIN, 2010; RAIZER; PARAENSE; GUDWIN, 2011; RAIZER; PARAENSE; GUDWIN, 2012; GUDWIN et al., 2013; PARAENSE; RAIZER; GUDWIN, 2016; PARAENSE et al., 2016; SIMOES; COLOMBINI; RIBEIRO, 2017) e *experimentos sensório-motores para a avaliação de agentes cognitivos* (BERTO, 2020). Além dessas abordagens, outras mais voltadas às aplicações em robôs podem ser destacadas: (SUN, 2004; SUN; FRANKLIN, 2007; LAIRD, 2008; FRANKLIN, 2012; LAIRD, 2012; FRANKLIN et al., 2016; MININGER; LAIRD, 2016).

1.1 Objetivos

1.1.1 Objetivo geral

Este trabalho tem como objetivo implementar e avaliar um conjunto de experimentos incrementais inspirados no desenvolvimento humano sobre uma arquitetura cognitiva para um robô simulado no contexto da *devRobotics*. A teoria do desenvolvimento cognitivo de Piaget e a literatura relacionada, em particular os experimentos propostos por Berto (2020), serão consideradas como guias para a realização dos experimentos com os agentes desenvolvidos e a validação dos módulos implementados.

1.1.2 Objetivos específicos

- Propor implementações computacionais baseadas nos experimentos sensório-motores de **Rastreamento de Objetos**, delineados em Berto (2020), compreendidos entre o 1º e 3º subestágio de desenvolvimento;
- Modelar e implementar em software (classes e subclasses) os módulos cognitivos do modelo CONAIM (*Conscious Attention-Based Integrated Model*) (SIMÕES, 2015) necessários para os experimentos supracitados, utilizando as ferramentas da computação cognitiva disponibilizadas pelo CST (*Cognitive System Toolkit*) (GUDWIN et al., 2013; PARAENSE; RAIZER; GUDWIN, 2016; PARAENSE et al., 2016);
- Implementar um único mecanismo de aprendizado procedimental, utilizando a arquitetura cognitivo-atencional CONAIM com as ferramentas cognitivas do *framework* CST, para demonstrar o processo incremental de aprendizado nos três primeiros subestágios sensório-motores da Teoria de Piaget;
- Executar os experimentos propostos com vistas a promover a investigação dos processos envolvidos no desenvolvimento cognitivo focados no domínio da robótica móvel, utilizando o ambiente CoppeliaSim (ROHMER; SINGH; FREESE, 2013).

1.2 Contribuições

Como principais contribuições deste trabalho, destacamos:

- O projeto e implementação do sistema top-down de atenção proposto pelo CONAIM no CST;
- A concepção e implementação de um modelo de aprendizado procedimental incremental;
- A concepção e implementação de uma máquina de processamento de esquemas;
- A avaliação da aprendizagem de um agente robótico através de experimentos cognitivos incrementais, inspirados na teoria de Piaget.

1.3 Organização do Texto

Este trabalho encontra-se organizado da seguinte forma: o Capítulo 2 destaca alguns dos modelos computacionais de arquiteturas cognitivas correlatos ao proposto. O Capítulo 3 destaca os principais conceitos relacionados à robótica do desenvolvimento (*devRobotics*). O Capítulo 4 apresenta as metodologias utilizadas no desenvolvimento deste trabalho e os experimentos propostos. O Capítulo 5 apresenta os resultados obtidos com os experimentos propostos. Finalmente, o Capítulo 5 apresenta a conclusão deste trabalho.

2 ARQUITETURAS E ESTRUTURAS COGNITIVAS

"O que é 'real'? Como você define o 'real'? Se você está falando sobre o que você pode sentir, o que você pode cheirar, o que você pode saborear e ver, o 'real' são apenas sinais elétricos interpretados pelo seu cérebro."

(Morpheus)
Matrix (1999)

A revolução cognitiva (HARARI, 2011) inspirou a investigação computacional dos aspectos envolvidos nos processos mentais humanos, que foram modelados em **arquiteturas e estruturas cognitivas**. Estas arquiteturas e estruturas buscam facilitar o estudo da mente humana, modelando não apenas seu comportamento, mas também os processos cognitivos subjacentes derivados deste comportamento ou que o influenciam (HARRIS, 2006). Nas últimas décadas, foram propostos modelos cognitivos com métodos distintos de funcionalidade e implementação (REGGIA, 2013; SIMÕES, 2015; KOTSERUBA; TSOTSOS, 2020a). Esta seção busca apresentar as principais contribuições da revolução cognitiva para o aprofundamento computacional nas ciências da mente, um histórico do desenvolvimento em modelos e arquiteturas cognitivas, uma taxonomia das arquiteturas, as principais características dos projetos envolvendo modelos e arquiteturas cognitivas, o modelo CONAIM, o CST Toolkit, a implementação CST+CONAIM e comparações desta implementação com arquiteturas cognitivas relevantes.

2.1 Revolução Cognitiva

A revolução cognitiva começou na década de 1950 como um estudo interdisciplinar da mente e de seus processos, com o objetivo de aplicar o método científico ao estudo da cognição humana através de experimentos computacionais para testar teorias sobre processos mentais humanos em ambientes controlados (HARARI, 2011; THAGARD, 2019; THOMAS, 2019). Uma visão derivada do funcionalismo, o **computacionalismo**, foi proposta para defender que máquinas conscientes, dotadas de mentes, pudessem ser criadas (SEARLE, 1993; SIMÕES, 2015). Metaforicamente, Reggia (2013), faz uma analogia entre os estados funcionais do *hardware* de um computador que executa um algoritmo e os estados mentais do cérebro.

O desenvolvimento de arquiteturas que possam aprender de forma autônoma, representar o ambiente internamente e desenvolver mecanismos cognitivos derivados da mente humana se difere dos projetos da área de inteligência artificial, em que objetivos pré-definidos motivam o robô a desenvolver tarefas simples (MEEDEN; BLANK, 2006).

Segundo Russel e Norvig (1995) e Franklin (2007), um **agente** pode perceber o ambiente através de sensores, modificar este ambiente através de atuadores, suas configurações iniciais devem refletir apenas conhecimentos prévios do ambiente e serem modificadas através de suas experiências adquiridas. Já um **agente inteligente** ou **racional** pode avaliar as consequências de suas ações através de seus estados internos do ambiente, desenvolver aprendizado e direcionar sua tomada de decisões para maximizar sua performance (RUSSEL; NORVIG, 1995; SUTTON; BARTO, 1998; FRANKLIN, 2007).

Os primeiros sistemas e modelos propostos na revolução cognitiva seguiam a abordagem clássica **cognitivista**, inspirada em símbolos computacionais para o estudo do conhecimento no cérebro. A abordagem cognitivista supõe que a consciência é formada por representações internas análogas às estruturas de dados de um computador e que os procedimentos relacionados a estas representações são semelhantes aos algoritmos computacionais (THAGARD, 2019). Estas representações contêm proposições lógicas, regras, conceitos, analogias, procedimentos mentais (como dedução, pesquisa, correspondência e recuperação) e armazenam partes separadas de informações em diferentes registros de

memória (KOTSERUBA; TSOTSOS, 2020a). As informações são acessadas com base nos registros de seus endereços numéricos e são dependentes da aplicação de regras sensíveis à sintaxe.

Com base em novas propostas sobre as possíveis representação do conhecimento, o **conexionismo** surgiu como um substituto da abordagem clássica (THAGARD, 2019). O conexionismo representa o conhecimento interno em estruturas análogas às redes neurais, inspiradas em conceitos como neurônios, suas conexões, ativação de disparo e propagação. Os detalhes específicos do funcionamento neural são extraídos para processos cognitivos de alto nível (KOTSERUBA; TSOTSOS, 2020a).

Conexionismo

2.2 Arquiteturas Cognitivas

De acordo com Kotseruba e Tsotsos (2020a), arquiteturas cognitivas são sistemas que podem raciocinar em diferentes domínios, desenvolver diferentes visões, adaptar-se a novas situações e refletir sobre si mesmas. Conforme relatam os autores, o termo “arquitetura cognitiva” não é restritivo, sendo aplicado a propostas de representações mentais e procedimentos computacionais que operam nessas representações. Para Paraense et al. (2016), arquiteturas cognitivas são sistemas de controle geral inspiradas em teorias científicas, desenvolvidas para explicar a cognição em humanos e outros animais, que compreendem composições de módulos responsáveis pela implementação de diferentes capacidades cognitivas, como percepção, atenção, memória, raciocínio e aprendizado.

Segundo Vernon, Hofsten e Fadiga (2016), uma arquitetura cognitiva desempenha o papel da cognição na modelagem computacional, tornando explícito o conjunto de processos e suposições que esse modelo cognitivo se baseia. Já para Laird (2008), uma arquitetura cognitiva consiste em unidades de processamento que representam, extraem, selecionam e combinam conhecimento e memórias que armazenam este conhecimento. As arquiteturas cognitivas são definidas por Sun (2004) como estruturas e processos essenciais com escopo amplo da cognição e do comportamento em vários níveis e múltiplos domínios, derivados de fontes biológicas ou psicológicas, argumentos filosóficos ou hipóteses de trabalho em diferentes disciplinas, como neurofisiologia, psicologia ou inteligência artificial. Para Thomson, Bennati e Lebiere (2014), arquiteturas cognitivas são sistemas que consistem em diversos módulos ou componentes que operam em conjunto para produzir comportamento. Estes módulos, segundo Anderson e Lebiere (1998), contêm representações de conhecimento, memórias para armazenamento de conteúdo e processos que adquirem e utilizam conhecimento.

2.3 Histórico e Taxonomia

Uma lista de arquiteturas cognitivas e suas características é apresentada em Samsonovich (2010a) e utilizada por Samsonovich (2010b) em uma revisão bibliográfica sobre a área. Uma linha do tempo de 87 estruturas e arquiteturas cognitivas é apresentada por Kotseruba e Tsotsos (2020a) e ilustrada na Figura 2. A bibliografia destas arquiteturas está disponível em Kotseruba e Tsotsos (2020d) e os links para os códigos-fonte de algumas destas arquiteturas está disponível em Kotseruba e Tsotsos (2020h). As arquiteturas são classificadas de acordo com seu tipo de representação do conhecimento: simbólicas (cognitivistas), emergentes (conexionistas) e híbridas.

As arquiteturas **simbólicas** ou **cognitivistas** possuem sua base de conhecimento representada a partir de símbolos (*labels*, *strings* de caracteres, *frames*), que podem ser manipulados por um conjunto de instruções (KOTSERUBA; TSOTSOS, 2020a). Este tipo de representação é ilustrado na Figura 1 (a). Sistemas simbólicos são menos capazes de lidar com a flexibilidade e a robustez necessárias para o processamento perceptivo em um ambiente instável, o que pode dificultar adaptação e aprendizado (VERNON; HOFSTEN; FADIGA, 2016; KOTSERUBA; TSOTSOS, 2020a). Além disso, uma arquitetura simbólica exige muito trabalho para criar sua base de conhecimento inicial. Dentre

Arquiteturas simbólicas ou cognitivistas

as arquiteturas simbólicas mais conhecidas, podem-se citar ICARUS (LANGLEY; CHOI; TRIVEDI, 2008) e EPIC (*Explicitly Parallel Instruction Computing*) (KIERAS, 2004).

Arquiteturas
Emergentes ou
Conexionistas

As arquiteturas **emergentes** ou **conexionistas** possuem sua base de conhecimento representada como redes neurais (KOTSERUBA; TSOTSOS, 2020a). Estas redes são compostas por componentes interativos simultâneos e distribuídos, massivamente paralelos, onde o fluxo de informações é representado por uma propagação de sinais dos nós de entrada (HAYKIN, 1998). O conhecimento não é um conjunto de entidades simbólicas, mas uma distribuição de redes (KRICHMAR et al., 2004). Este tipo de representação é ilustrado na Figura 1 (b). As abordagens emergentes concentram-se no desenvolvimento do agente de um estado primitivo para um estado cognitivo em um longo período de tempo (VERNON; HOFSTEN; FADIGA, 2016). O termo "arquitetura cognitiva" é utilizado para denotar a estrutura que facilita esse desenvolvimento. Em contraste com a postura cognitivista, a presença de capacidades inatas em um sistema emergente não implica necessariamente que a arquitetura seja funcionalmente modular (KOTSERUBA; TSOTSOS, 2020a). Dentre as arquiteturas emergentes mais conhecidas, podem-se citar BBD (*Brain-Based Device*) (KRICHMAR et al., 2004), Subsumption (BROOKS, 1985), SPA (*Semantic Pointer Architecture*) (STEWART; BLOUW; ELIAS-MITH, 2015), SASE (*Self-Affecting Self-Aware*) (WENG; LEE; EVANS, 1999) e MDB (*Multilevel Darwinist Brain*) (BELLAS et al., 2010).

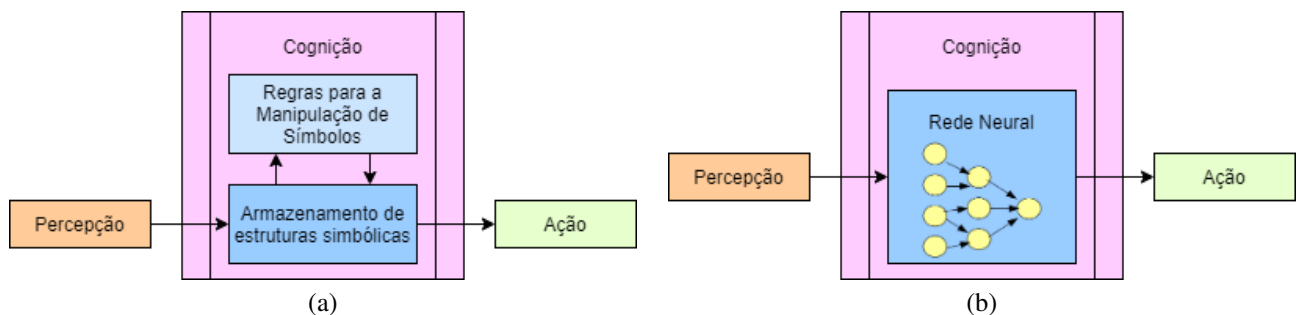


Figura 1 – a) Ilustração do processo de percepção, representação e ação em um sistema **cognitivista**. b) Ilustração do processo de percepção, representação e ação em um sistema **conexionista**. Fonte: (GUDWIN, 2014) (adaptado).

Arquiteturas
Híbridas

As arquiteturas **híbridas** propõem combinar elementos de abordagens simbólicas (cognitivistas) e emergentes (conexionistas). São utilizadas diferentes abordagens de hibridização e combinação de representações nas arquiteturas e estruturas mais conhecidas. As arquiteturas ACT-R (*Adaptive Control of Thought - Rational*) (ANDERSON; LEBIERE, 1998), Copycat (HOFSTADER; MITCHELL, 1995), SOAR (*State, Operator, and Result*) (LAIRD, 2008; LAIRD, 2012; MININGER; LAIRD, 2016) e CLARION (*Connectionist Learning with Adaptive Rule Induction On-line*) (SUN, 1999; SUN, 2004; SUN; FRANKLIN, 2007) combinam conceitos simbólicos e regras com elementos sub-simbólicos, como valores de ativação, propagação de ativação e Aprendizado por Reforço. Outras arquiteturas, como LIDA (*Learning Intelligent Distribution Agent*) (FRANKLIN, 2012), iCub (VERNON; METTA; SANDINI, 2007) e FORR (*FOR the Right Reasons*) (EPSTEIN, 2001), são implementadas como um conjunto de módulos concorrentes e cooperativos interconectados, em que módulos individuais não estão restritos a uma determinada representação. Outra estratégia de hibridização é combinar duas arquiteturas independentes diferentes com recursos complementares. Na arquitetura robótica ADAPT (*Adaptive Dynamics and Active Perception for Thought*) (BENJAMIN; LYONS; LONSDALE, 2004), a arquitetura SOAR é utilizada para controle, enquanto os módulos separados são responsáveis por modelar um mundo 3D a partir das informações do sensor de visão e pelo processamento visual.

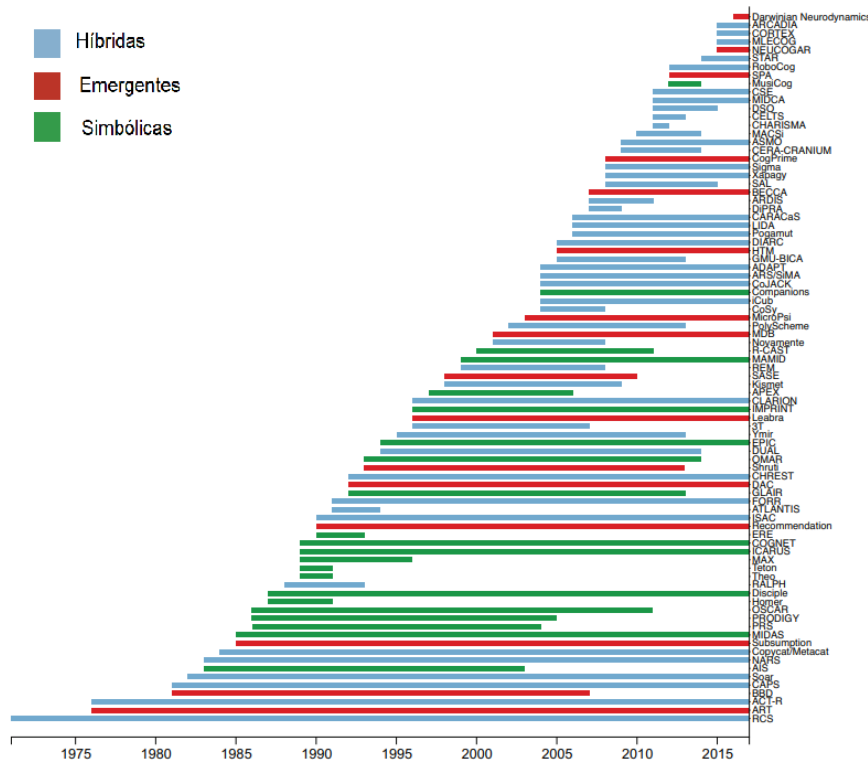


Figura 2 – Cronologia de 84 arquiteturas e estruturas cognitivas baseadas em representação da mente e processamento cognitivo. As arquiteturas são classificadas em **simbólicas**, **emergentes** e **híbridas**. As barras determinam o início do desenvolvimento dos modelos e se foram descontinuados. Fonte: (KOTSERUBA; TSOTSOS, 2020a) (adaptado).

2.4 Características

Um mapeamento de trabalhos envolvendo mecanismos cognitivos utilizados nas arquiteturas cognitivas é apresentado por Kotseruba e Tsotsos (2020c). As arquiteturas utilizadas nestes trabalhos são avaliadas quanto a existência dos mecanismos cognitivos de percepção, atenção, tomada de decisão (ou seleção de ações), memórias e aprendizado.

2.4.1 Percepção e Sensoriamento

Um sistema inteligente pode processar diversos tipos e quantidades de dados disponíveis no ambiente através de entradas sensoriais perceptivas (KRICHMAR et al., 2004). Um mapeamento de modalidades sensitivas já implementados em arquiteturas e estruturas cognitivas é apresentado por Kotseruba e Tsotsos (2020g). Nota-se que a visão é a modalidade sensorial mais implementada e mais da metade das arquiteturas utilizam de simulações como entrada visual. Implementações como toque e cinestesia são utilizadas principalmente nas arquiteturas fisicamente incorporadas (WENG; JOHANNES, 2006).

2.4.2 Memórias

Nas implementações de arquiteturas cognitivas, a memória é descrita em termos de duração (curto e longo prazo) e tipo (procedimental, episódico, semântico), embora não seja necessariamente implementada como armazenamento de conhecimentos separados. As memórias são caracterizadas de acordo com seu tipo de armazenamento, a curto ou longo prazo (SIMÕES; COLOMBINI; RIBEIRO, 2016). Projetos que envolvem planejamento geralmente representam

o armazenamento de curto prazo em um modelo do mundo atual ou como o conteúdo da pilha de objetivos. Um mapeamento de modalidades de memórias já implementados em arquiteturas cognitivas é apresentado por [Kotseruba e Tsotsos \(2020f\)](#).

Memórias de Curto Prazo As **memórias de curto prazo** são limitadas em tamanho e armazenam dados obtidos da interação dos indivíduos com o ambiente, com objetos ou com outros indivíduos, em tempo real ([SIMÕES, 2015](#)). As memórias de curto prazo podem ser subdivididas em: memória sensorial, memória de trabalho e memória motora.

Memória sensorial • **Memória sensorial:** permite reter impressões de informação sensorial após o estímulo original ter cessado ([PRATAMA; MASTROGIOVANNI; CHONG, 2014](#));

Memória de trabalho • **Memória de trabalho:** possui capacidade limitada, representa mentalmente as situações atuais e armazena informações para o raciocínio em andamento ([SIMÕES, 2015](#));

Memória motora • **Memória motora:** memória relacionada às possíveis ações do sistema e em como atuadores são manipulados para executar estas ações ([PRATAMA; MASTROGIOVANNI; CHONG, 2014](#)).

Memórias de longo prazo As **memórias de longo prazo** armazenam o conhecimento adquirido durante todo o tempo de vida de um indivíduo ([SIMÕES, 2015](#)). As informações são obtidas pelas memórias de longo prazo após a análise dos dados reais armazenados nas memórias de curto prazo. Podem ser subdivididas em: memória perceptual, memória semântica, memória episódica e memória procedimental.

Memória perceptual • **Memória perceptual:** memória que interpreta os padrões de estímulos recebidos e auxilia no reconhecimento, categorização e relacionamento de ambientes, indivíduos ou objetos ([FRANKLIN, 2012](#));

Memória semântica • **Memória semântica:** memória que armazena fatos e seus significados, conceitos gerais e abstratos, conhecimentos do senso comum, não contextualizados no tempo e no espaço ([TULVING, 1985](#); [SIMÕES, 2015](#); [PRATAMA; MASTROGIOVANNI; CHONG, 2014](#)). Esta memória é intrinsecamente relacionada à memória perceptual;

Memória episódica • **Memória episódica:** memória que se refere a eventos específicos (o que, quando e onde), localizados no tempo e espaço ([PRATAMA; MASTROGIOVANNI; CHONG, 2014](#)). Estes eventos são associados com detalhes particulares. Conforme destaca [Simões \(2015\)](#), a memória episódica influencia a memória semântica, uma vez que a experiência adquirida em situações específicas realimenta os dados semânticos;

memória procedimental • **Memória procedimental:** memória responsável pelo aprendizado de funções, também conhecidas como habilidades motoras e comportamentais. A memória procedimental permite que conexões aprendidas entre estímulos e respostas sejam retidas, incluindo aquelas que envolvem padrões complexos de estímulos e respostas adaptativas ao ambiente ([TULVING, 1985](#)). Esta memória recebe informações da memória motora e armazena informações sobre como executar certos procedimentos ([FRANKLIN, 2012](#)).

As memórias também podem ser classificadas de acordo com suas possíveis formas de armazenamento e tipos de informações ([SQUIRE, 2004](#)). As **memórias declarativas** são conscientemente acessadas, facilmente verbalizadas e com seu conteúdo descrito na linguagem corrente (Ex: memórias semântica e episódica). As **memórias não-declarativas** caracterizam-se como coleções heterogêneas de capacidades inconscientes e seu acesso é expresso pelo desempenho e interação das informações em tarefas anteriores ao invés da recordação de conceitos definidos (Ex: memórias perceptual e procedimental) ([SQUIRE, 1992](#); [SQUIRE, 2004](#)).

Memória declarativa
Memória não-declarativa

2.4.3 Atenção

A atenção seleciona informações relevantes e filtra informações irrelevantes dos dados sensoriais (COLOMBINI, 2014). A atenção pode ser modelada como um conjunto de mecanismos que afetam os processos perceptivos e cognitivos (VERNON; METTA; SANDINI, 2007). Por exemplo, ao combinar informações sensoriais em mapas de características, um sistema pode identificar os parâmetros mais adequados a seus objetivos e direcionar seu foco de atenção. De acordo com Tsotsos (2011), os elementos de atenção podem ser agrupados em três classes de mecanismos de redução de informações: seleção, restrição e supressão. Inspirados na taxonomia de mecanismos atencionais proposta por Tsotsos (2011), Kotseruba e Tsotsos (2020i) apresentam um mapeamento de modalidades de atenção visual já implementados nas arquiteturas cognitivas.

Um método para quantificar o processo atencional e seus mecanismos foi proposto por Koch e Ullman (1985) em um mapa topograficamente orientado, um **mapa de saliências**. Neste mapa, as diferentes informações que contribuem para a seleção de um estímulo (como distância, orientação, cor, etc.) são normalizadas e integradas em medidas globais com o mapa de atenção do agente (KOCH; ULLMAN, 1985; COLOMBINI; SIMÕES; RIBEIRO, 2017). De acordo com Koch e Ullman (1985), o mapa de saliências foi projetado como uma entrada para mecanismos de controle para a atenção seletiva e que a localização mais saliente do mapa seria um bom candidato à seleção atencional.

Mapa de
saliências

- **Mecanismos de seleção:** apenas uma característica ou objeto de interesse é selecionado dentre muitos (VERNON; METTA; SANDINI, 2007). Estes mecanismos incluem: **i) seleção a partir de modelo global:** objetos ou eventos já conhecidos são utilizados para focar; e **ii) seleção a partir de itens de interesse:** seleção a partir de tempo, região, características, objetos ou eventos (ANDERSON; LEBIERE, 1998);
- **Mecanismos de restrição:** são selecionadas algumas características ou objetos de interesse dentre muitos (COLOMBINI, 2014). Estes mecanismos incluem *i) Motivações endógenas:* estímulos internos motivam a seleção. São implementadas nas arquiteturas como instruções de tarefas (LUNGARELLA; SANDINI; PFEIFER, 2003); e *ii) Motivações exógenas:* estímulos externos motivam a seleção. São implementadas como conhecimento do domínio. A atenção também pode ser modulada por sinais internos, como motivação e emoções;
- **Mecanismos de supressão:** são excluídas algumas características ou objetos de interesse dentre muitos (WANG; MILLER; LIU, 2015). Um exemplo deste tipo de mecanismo é a **inibição de retorno de localização ou objeto (IOR)**, ou seja, a exclusão ou diminuição temporária de características ou do ambiente espacial durante a execução, de forma a não voltar a atenção para localizações ou estímulos previamente assistidos (NYAMSUREN; TAATGEN, 2013).

Um modelo atencional computacional foi proposto por Colombini (2014), em que a atenção é vista como a condição de prontidão que seletivamente foca consciência e receptividade dos estímulos e direciona a aplicação da mente a um determinado objeto ou pensamento. De acordo com Colombini (2014), um processo atencional pode ser disparado de duas formas distintas: atenção exógena e atenção endógena. Os processos de **atenção exógena** ou *bottom-up* são iniciados involuntariamente, os estímulos são apresentados de forma inesperada e qualquer tipo de estímulo pode despertar a atenção. Os processos de **atenção endógena** ou *top-down* são voluntários, controlados e são iniciados por processos cognitivos ou psicológicos através de uma mobilização consciente da atenção (POSNER, 1980; COLOMBINI, 2014).

bottom-up

top-down

2.4.4 Tomada de decisão

A **tomada de decisão** ou **seleção de ações** se refere à realização de tarefas ou de comandos motores. Neste processo, é definido o que um agente fará em seguida, com base em suas entradas, ações disponíveis e seus objetivos (PAISNER et al., 2013; COLOMBINI; SIMÕES; RIBEIRO, 2017; KOTSERUBA; TSOTSOS, 2020a). Um mapeamento dos

Tomada de
decisão

mecanismos de tomada de decisão e seleção de ações já implementados em arquiteturas cognitivas é apresentado por Kotseruba e Tsotsos (2020b). Os mecanismos de seleção de ações são implementados, em sua maioria, como *winner-takes-all*.

- **Winner-takes-all (WTA):** Processo de seleção onde o conjunto mais forte de entradas é amplificado e escolhido, enquanto demais conjuntos são inibidos (FRANKLIN, 2012; COLOMBINI; SIMÕES; RIBEIRO, 2017);
- **Escolhas por relevância:** A seleção é feita de acordo com a correspondência das ações à situação atual, através de medidas e condições das contribuições esperadas de cada ação para a aplicação no processo de seleção (SUN, 2004; LAIRD, 2012);
- **Escolhas reativas:** Decisões espontâneas, simples e rápidas, respostas automáticas a estímulos, com ações executadas imediatamente, suspendendo qualquer atividade em andamento e ignorando o raciocínio (BROOKS, 2003; COLOMBINI; SIMÕES; RIBEIRO, 2017).

2.4.5 Aprendizado

O aprendizado é a capacidade de um sistema em melhorar seu desempenho ao longo do tempo com a aquisição ou aperfeiçoamento de comportamentos (SQUIRE, 1992). Um sistema inteligente pode ser capaz de inferir fatos e comportamentos a partir dos eventos observados ou dos resultados de suas próprias ações, avaliar seu progresso, lidar com restrições em tempo real, possíveis dados incorretos dos sensores e atrasos de detecção e atuação (SQUIRE, 1992; KOTSERUBA; TSOTSOS, 2020a). Além de aprender as tarefas básicas de controle, um algoritmo de aprendizado deve aprender a compensar os atrasos na detecção e na atuação e prever o comportamento em seu ambiente. O aprendizado por reforço (*Reinforcement Learning* - RL) utiliza diferenças temporais para avaliar consequências de ações como recompensas (SUTTON; BARTO, 1998). Um agente tende a repetir uma resposta que foi reforçada por consequências desejáveis, aumentando a probabilidade de que seja repetido, e a suprimir uma resposta que foi punida, diminuindo a probabilidade de repetição.

Os ambientes em algoritmos de aprendizado normalmente são representados em vetores de estados que possuem a propriedade de Markov. Cada estado representa estatisticamente todas as informações relevantes do agente até determinado instante (RUSSEL; NORVIG, 1995; SUTTON; BARTO, 1998). Um **processo de decisão de Markov** (MDP - *Markov Decision Process*) é caracterizado como o processo de decisão em um ambiente completamente observável e estocástico, que utiliza de estados com a propriedade de Markov e recompensas (RUSSEL; NORVIG, 1995; SUTTON; BARTO, 1998). Se os espaços de estados e ações são finitos, dados estado atual e ação selecionada, pode ser definida a probabilidade de novos estados e recompensas (RUSSEL; NORVIG, 1995). A cada etapa do processo, o agente seleciona uma ação com base no estado atual. As consequências da ação selecionada são avaliadas de acordo com o desempenho deste agente e são definidas como recompensas. Uma distribuição de probabilidade entre ações, dados os estados, é denominada como política, e define o comportamento de um agente e sua tomada de decisões (SUTTON; BARTO, 1998). Uma política que maximiza as recompensas recebidas é denominada política ótima (RUSSEL; NORVIG, 1995).

Os algoritmos de RL podem ser baseados em modelo e utilizar as funções de transição e recompensas para estimar a política ideal, ou livres de modelo e estimar a política ideal sem funções de transição e recompensas do ambiente.

Um algoritmo de RL, livre de modelo, derivado de **processos de Decisão de Markov**, é o **QLearning**, um processo assíncrono de programação dinâmica (WATKINS; DAYAN, 1992; SUTTON; BARTO, 1998). Para qualquer MDP finito, a partir do estado atual, o Qlearning encontra uma política ideal para maximizar o valor esperado de recompensa em todas as etapas sucessivas (WATKINS; DAYAN, 1992; RUSSEL; NORVIG, 1995; SUTTON; BARTO, 1998). O

Aprendizado
por reforço

QLearning

mecanismo atualiza a qualidade / utilidade de uma determinada ação em um determinado estado de acordo com a recompensa para esta ação (WATKINS; DAYAN, 1992; SUTTON; BARTO, 1998).

$$Q_{new}(s_t, a_t) \leftarrow Q_{old}(s_t, a_t) + \alpha * (r_t + \gamma * \max_a Q(s_{t+1}, a) - Q_{old}(s_t, a_t)) \quad (1)$$

Onde:

- $Q_{new}(s_t, a_t)$ é o valor atualizado para a qualidade / utilidade da ação a_t no estado s_t ;
- $Q_{old}(s_t, a_t)$ é o valor anterior à atualização para a qualidade / utilidade da ação a_t no estado s_t ;
- α é a taxa de aprendizado, que determina em que medida informações recém-adquiridas substituem informações antigas;
- r_t é o valor da recompensa para a ação a_t no estado s_t ;
- γ é o fator de desconto, que determina a importância de recompensas futuras;
- $\max_a Q(s_{t+1}, a)$ é o valor máximo de recompensa em um novo estado s_{t+1} , considerando todas as ações A .

Um mapeamento das aplicações de diferentes tipos de aprendizado em arquiteturas cognitivas é apresentado por Kotseruba e Tsotsos (2020e). Os mecanismos envolvendo aprendizado em máquinas podem ser divididos pelo tipo de aquisição do conhecimento:

- **Declarativo:** Aprendizado com base em coleções de fatos sobre o mundo e relacionamentos entre eles (SUN, 2004);
- **Perceptivo:** Aprendizado utilizado para obter conhecimento explícito sobre o ambiente e aplicado em arquiteturas que mudam ativamente como informações sensoriais são tratadas ou como padrões são aprendidos (KOTSERUBA; TSOTSOS, 2020a);
- **Procedimental:** Aprendizado de comportamentos que acontecem momentaneamente e gradualmente, através da repetição, se tornam automáticos (HAYES-ROTH et al., 1993). Grande parte das implementações de aprendizado procedimental é realizada como **Aprendizado por reforço**, onde processos de tomada de decisão são influenciados por recompensa (VERNON; METTA; SANDINI, 2007).

2.5 CONAIM

O modelo CONAIM (*Conscious Attention-Based Integrated Model*) (SIMOES; COLOMBINI; RIBEIRO, 2017) foi inspirado sobre a arquitetura atencional proposta por Colombini (2014), em substituição a seu controlador atencional. O modelo é capaz de dotar um agente baseado em atenção de consciência e de computar sobre o mapa de saliências, reduzindo assim de forma significativa o espaço de dimensões de entrada do modelo. Compreende dois sistemas principais: um atencional e um cognitivo. Sua estrutura é demonstrada na Figura 5.

2.5.1 Sistema atencional

O sistema atencional do modelo proposto por Simões (2015) foi baseado no modelo de **seleção para percepção** proposto por Colombini (2014). Esta arquitetura incorpora diversos aspectos de outros trabalhos correlatos e é capaz de lidar com múltiplos sistemas sensoriais, múltiplos processos para extração de características, processo decisório e suporte a aprendizado (COLOMBINI; SIMÕES; RIBEIRO, 2017).

Durante o ciclo *bottom-up*, sensores fornecem dados externos armazenadas em uma janela de tempo na memória sensorial para formar mapas de características (COLOMBINI; SIMÕES; RIBEIRO, 2017). Estes mapas são combinados em um único mapa de características combinado, que representa uma soma ponderada dos recursos. A junção do mapa de características combinado com o mapa atencional gerado anteriormente pelo agente gera um mapas de saliências, uma representação do ambiente percebido pelo agente, modulado pelo processo atencional. Através de uma seleção de vencedores, por exemplo, por um algoritmo *winner-takes-all*, pode-se definir o estímulo mais relevante no mapa de saliências. O modelo também fornece uma sistemática de inibição (IOR), que promove o decaimento da sensibilidade de uma região quando existe uma exibição prolongada do estímulo, promovendo a atenção para o próximo local com o segundo valor mais alto no mapa de saliência. A sucessão desses eventos gera uma varredura sequencial da cena.

Da perspectiva *bottom-up* apenas a região mais saliente da combinação de recursos terá a atenção direcionada. Quanto à cronometria do processo, essa característica ou região estimulará a si mesma e ao seu redor após um tempo t para outro $t + y$. Depois disso, o efeito de inibição de retorno de localização ou objeto comprimirá a região previamente aprimorada por outro período de tempo. Esta modelagem é baseada nas gaussianas ilustradas a Figura 4. O modelo considera as contribuições máximas e mínimas da atenção perdidas com o tempo (Equação 4, demonstrada abaixo), com base em uma curva usual adotada para modelar a integração da atividade neuronal, e também descreve a influência excitatória e inibitória que será demonstrada pela atenção nas regiões vizinhas da região do Vencedor (Equação 5). Um processo para a inibição de uma região mantida por muito tempo é modelado na Equação 6, prejudicando os estímulos no local frequentado, permitindo que a orientação da atenção se afaste da manutenção da meta (COLOMBINI; SIMÕES; RIBEIRO, 2017).

Na interação *top-down* o sistema recebe alterações de um bloco cognitivo de individualidade através do estado de atatividade global e de outro bloco cognitivo de tomada de decisão, interferindo nos atuadores, alterando o ambiente, e assim novamente todo o sistema. Este processo depende do objetivo do sistema e da dinâmica atencional do estado corrente: orientada, seletiva ou sustentada (COLOMBINI, 2014). A **atenção orientada** é resultado do processo atencional para escolher o foco de atenção. A **atenção seletiva** pode enfatizar características específicas. A **atenção sustentada** é guiada pelo objetivo. De acordo com Colombini (2014), em sistemas biológicos, o componente endógeno da atenção inicia 200ms depois que o estímulo é atendido, ele permanece enquanto a tarefa o fizer necessário.

A Figura 3 apresenta a arquitetura de Colombini e os elementos envolvidos nos estímulos *bottom-up* e *top-down*, que competem pelo foco da atenção.

Formalmente, o modelo é descrito como:

- Um conjunto de sensores $\{s_1, s_2, \dots, s_{ns}\} \in \mathcal{S}$;
- Um conjunto de dimensões de sensores $\{sd_1, sd_2, \dots, sd_{ns}\} \in \mathcal{S}d$, com cada $sd_i \in \mathcal{S}d$ associada ao sensor $s_i \in \mathcal{S}$;
- Um conjunto de observações dos sensores, denotado por $\{\mathcal{O}_1, \mathcal{O}_2, \dots, \mathcal{O}_i\} \in \mathcal{O}$, onde $\mathcal{O}_{i_jk}$ representa a observação da dimensão j , do sensor i , no tempo k , $k \in [t - k, t]$, e t é o instante da observação. i: sensor (Ex: Sensor de visão); j: dimensão (Ex: Pixels da imagem recebida); k: tempo.
- Um número de dimensões características $d \in \mathbb{N}^*$ nas quais características *bottom-up* podem emergir ou características *top-down* podem atuar;
- Um conjunto de d funções de dimensões características *bottom-up* $\{\phi_{BU_1}, \phi_{BU_2}, \dots, \phi_{BU_d}\} \in \Phi_{BU}$;
- Um conjunto de d funções de dimensões características *top-down* $\{\phi_{TD_1}, \phi_{TD_2}, \dots, \phi_{TD_d}\} \in \Phi_{TD}$;
- Um conjunto de $2d$ mapas de características *bottom-up* n -dimensionais $\mathcal{F}_{BU} \in \mathbb{R}^n$, com elementos denotados por $\{\mathcal{F}_{BU_1}, \mathcal{F}_{BU_2}, \dots, \mathcal{F}_{BU_d}\}$, com cada mapa de características $\mathcal{F}_{BU_i}$ representado por $\{f_{i_1}, f_{i_2}, \dots, f_{i_n}\}$;

- Um vetor de pesos de características d -dimensional $\mathcal{W}_f \in \mathbb{R}^d$, com elementos denotados por $\{wf_1, wf_2, \dots, wf_d\}$;
- Um mapa de características combinado n -dimensional $\mathcal{C} \in \mathbb{R}^n$, com elementos denotados por $\{c_1, c_2, \dots, c_n\}$;
- Um mapa de saliências n -dimensional $\mathcal{L} \in \mathbb{R}^n$, com elementos denotados por $\{l_1, l_2, \dots, l_n\}$;
- Um mapa atencional n -dimensional $\mathcal{M} \in \mathbb{R}^n$, com elementos denotados por $\{m_1, m_2, \dots, m_n\}$, todos inicializados com valor 1;
- Um conjunto de Vencedores $\mathcal{B}$, com elementos denotados por $\{b_1, b_2, \dots, b_z\}$. Cada elemento $b_i \in \mathcal{B}$ é uma 3-tupla $\{\xi, t, (0, 1)\}$ onde os elementos são: 1) Index ξ que refere-se ao vencedor $l_i \in \mathcal{L}$; 2) Tempo de ativação t ; e 3) um elemento binário $\in \{0, 1\}$ que indica se a ativação foi causada por um processo exógeno ou endógeno;
- Um conjunto de objetivos $\{g_1, g_2, \dots, g_t\} \in \mathcal{G}$;
- Uma função de tomada de decisão α que mapeia o mapa de saliências ($\mathcal{L}$) e o objetivo g_t de acordo com o Index do Vencedor;
- Uma função de mapeamento de pesos *top-down* ζ_{w_f} ;
- Uma função de mapeamento de dimensões características *top-down* $\zeta_{\phi_{TD}}$.

A operação do modelo é apresentada por [Colombini \(2014\)](#):

1. **Atualização *top-down*.** De acordo com as funções de mapeamento ζ_{w_f} e $\zeta_{\phi_{TD}}$, com base no objetivo g no tempo t , atualizar o vetor de pesos de características $\mathcal{W}_f$ e as funções de dimensão de características *top-down* ϕ_{TD} ;
2. **Extração de características.** Para todas as dimensões d , aplicar uma função de dimensão de características *bottom-up* ϕ_{BU_d} , capaz de extrair um mapa de características *bottom-up* n -dimensional denotado por $\mathcal{F}_{BU_d}$, e uma função de dimensão de características *top-down*, capaz de extrair um mapa de características *top-down* denotado por $\mathcal{F}_{TD_d}$, ambos no tempo t , estabelecendo os mapeamentos:

$$\mathcal{O} \xrightarrow{\phi_{BU_d}} \mathcal{F}_{BU_{d_t}} : \mathbb{R}^n$$

$$\mathcal{O} \xrightarrow{\phi_{TD_d}} \mathcal{F}_{TD_{d_t}} : \mathbb{R}^n$$

3. **Combinação dos mapas de características.** Para todo $i \in \{1, 2, \dots, n\}$, computar $c_{i_t} \in \mathcal{C}_t$:

$$c_{i_t} = \sum_{j=1}^d wf_{j_t} \cdot \mathcal{F}_{i_{j_t}}$$

4. **Computação do mapa de saliências.** Para todo $i \in \{1, 2, \dots, n\}$, computar $\mathcal{L}_{d_t}$, fazendo:

$$l_{i_t} = c_{i_t} \cdot m_{i_{t-1}},$$

com $m_{i_{t-1}} \in \mathcal{M}_{t-1}$ e $c_{i_t} \in \mathcal{C}_t$;

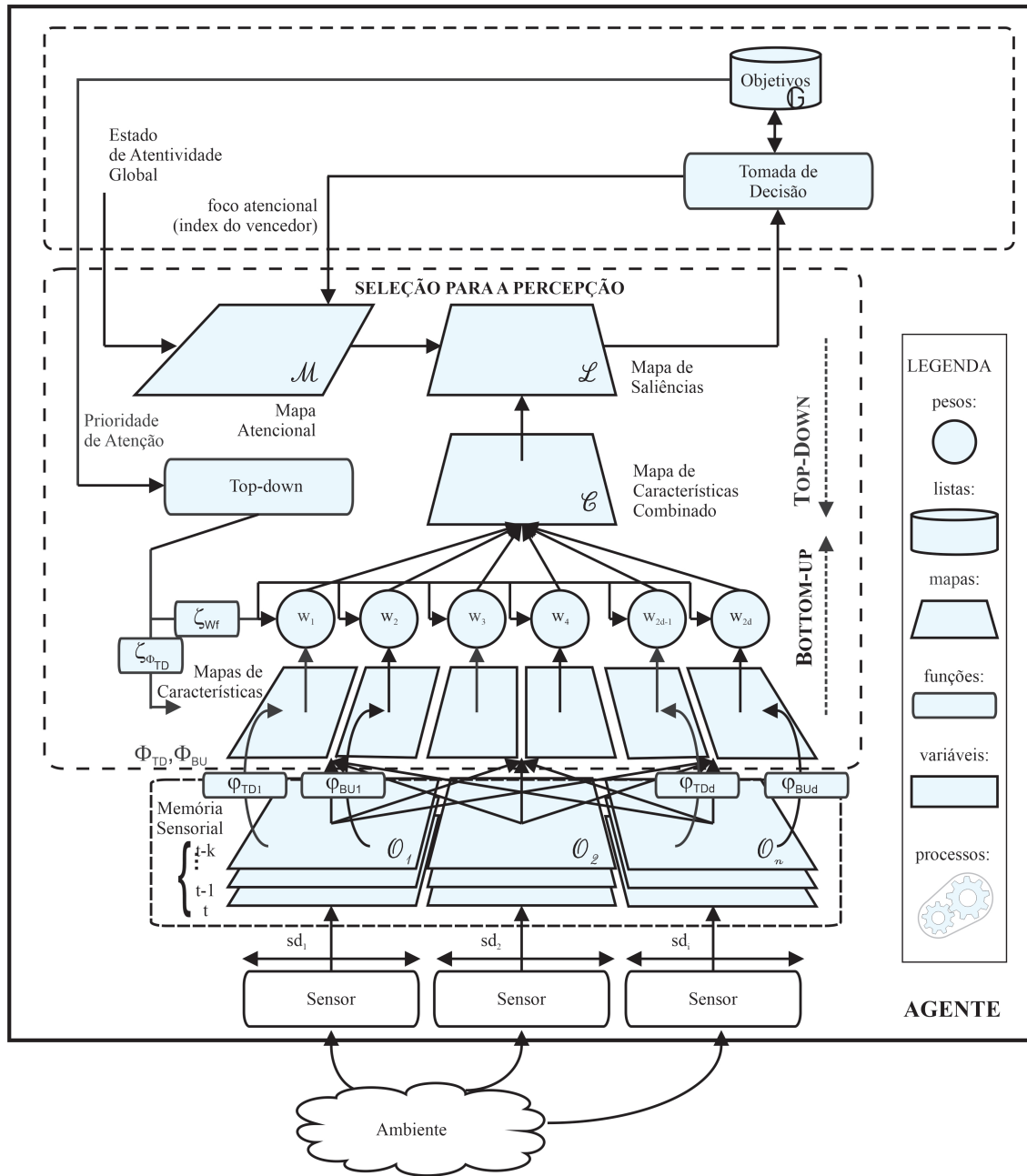


Figura 3 – Modelo atencional de Colombini (2014) Fonte: (COLOMBINI, 2014) (adaptado).

5. **Tomada de decisão.** Executar o mapeamento do mapa de saliências $\mathcal{L}$ e o objetivo g_t de acordo com o Index do Vencedor ξ . Inserir o Vencedor $b_{i(\xi, t, (0,1))}$ em $\mathcal{B}$. Com o processo de decisão *winner-takes-all* em curso, o Index do Vencedor ξ de b_i será computado sobre o mapa de saliências:

$$\xi = j, \text{ com } l_{j_t} = \max(\mathcal{L}), \text{ para } j \in \{1, 2, \dots, n\}$$

6. **Atualização do mapa atencional.**

- a) Para cada $b_i \in \mathcal{B}$ que ativou no tempo q , com $q \leq t$:

- i. Se b_i disparou com um estímulo exógeno (*bottom-up*), computar a contribuição $\Delta b_{i(\xi,q,0)}$ no instante t com respeito ao index do vencedor ξ fazendo:

$$\Delta b_{i(\xi,q,0)} = \left(\left[e^{\left(-\frac{t-q-\delta_p}{\tau_{aa}} \right)} - e^{\left(-\frac{t-q-\delta_p}{\tau_{ad}} \right)} \right] \cdot \mathcal{H}(t-q-\delta_p) \right) - \left(\left[e^{\left(-\frac{t-q-\delta_r}{\tau_{ra}} \right)} - e^{\left(-\frac{t-q-\delta_r}{\tau_{rd}} \right)} \right] \cdot \mathcal{H}(t-q-\delta_r) \right) \quad (2)$$

, onde:

$$\left\{ \begin{array}{l} \delta_p \text{ é um delay de pré-ativação constante;} \\ \delta_r \text{ é um delay de pré-refração constante;} \\ \tau_{aa} \text{ é a constante de ativação exponencial ascendente;} \\ \tau_{ad} \text{ é a constante de ativação exponencial descendente,} \\ \text{com } 0 < \tau_{ad} < \tau_{aa}; \\ \tau_{ra} \text{ é a constante de refração exponencial ascendente;} \\ \tau_{rd} \text{ é a constante de refração exponencial descendente,} \\ \text{com } 0 < \tau_{rd} < \tau_{ra}; \\ \xi \text{ é o componente do Index do Vencedor do elemento } b_i; \\ \mathcal{H} \text{ é a função degrau (função de Heaviside (SAUTER; AZEVEDO; STRAUCH, 2019)),} \\ \text{não definida para } t < 0, \text{ definida como } 1/2 \text{ para } t = 0 \text{ e } 1 \text{ para } t > 0; \end{array} \right.$$

- ii. Se b_i disparou com um estímulo exógeno (*bottom-up*), computar a contribuição $\Delta b_{i(j,q,0)}$ no instante t com respeito ao index do vencedor ξ e sua vizinhança. Para todo $j \in \{1, 2, \dots, n\}$ com $j \neq \xi$, fazer:

$$\Delta b_{i(j,q,0)} = \Delta b_{i(\xi,q,0)} \cdot e^{-\frac{(j-\xi)^2}{2\tau_\xi^2}} \quad (3)$$

, onde τ_ξ é uma constante relacionada à vizinhança.

- iii. Se b_i disparou com um estímulo endógeno (*top-down*), computar a contribuição $\Delta b_{i(\xi,q,1)}$ no instante t com respeito ao index do vencedor ξ , fazendo:

$$\Delta b_{i(\xi,q,1)} = \left(e^{\left(-\frac{t-q-\delta_{ep}}{\tau_{eaa}} \right)} \cdot \mathcal{H}(t-q-\delta_{ep}) \right) - \left(e^{\left(-\frac{t-q-\delta_{edp}}{\tau_{ead}} \right)} \cdot \mathcal{H}(t-q-\delta_{edp}) \right) \quad (4)$$

, onde:

$$\left\{ \begin{array}{l} \delta_{ep} \text{ é um delay de pré-ativação constante na excitação endógena;} \\ \delta_{edp} \text{ é um delay de pré-ativação constante no decaimento endógeno;} \\ \tau_{eaa} \text{ é a constante de ativação endógena exponencial ascendente;} \\ \tau_{ead} \text{ é a constante de ativação exponencial endógena descendente,} \\ \text{com } 0 < \tau_{ead} < \tau_{eaa}; \\ \mathcal{H} \text{ é a função degrau (função de Heaviside (SAUTER; AZEVEDO; STRAUCH, 2019)),} \\ \text{não definida para } t < 0, \text{ definida como } 1/2 \text{ para } t = 0 \text{ e } 1 \text{ para } t > 0; \end{array} \right.$$

- iv. Se b_i disparou com um estímulo endógeno (*top-down*), computar a contribuição $\Delta b_{i(j,q,1)}$ no instante t com respeito ao index do vencedor ξ e sua vizinhança. Para todo $j \in \{1, 2, \dots, n\}$ com $j \neq \xi$, fazer:

$$\Delta b_{i(j,q,1)} = \Delta b_{i(\xi,q,1)} \cdot e^{-\frac{(j-\xi)^2}{2\tau_{en}^2}} \quad (5)$$

, onde τ_{en} é uma constante endógena relacionada com a janela de vizinhança.

b) Para todo $m_j \in \mathcal{M}$, atualizar m_{j_t} com $j \in \{1, 2, \dots, n\}$ fazendo:

$$m_{j_t} = m_{j_{t-1}} + \Delta b_{i(j,q,0)} + \Delta b_{i(j,q,1)} \quad (6)$$

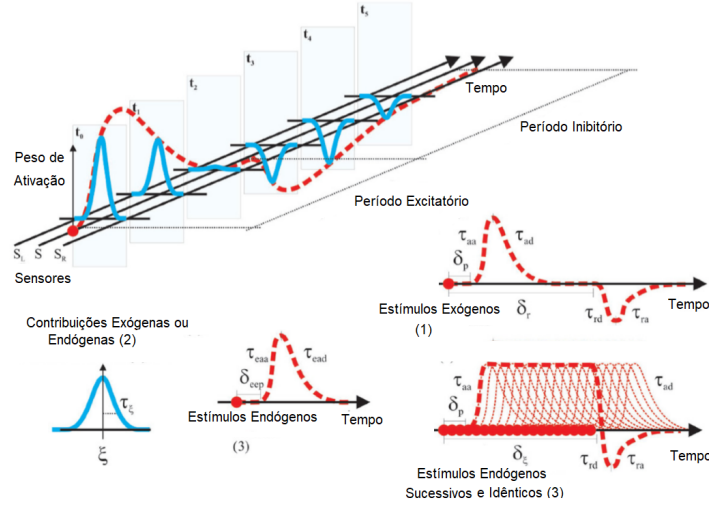


Figura 4 – Contribuição no aumento e decaimento de estímulos atencionais ao longo do tempo. Curvas típicas para: contribuições laterais endógenas e exógenas; estímulos exógenos (com IOR); estímulos endógenos (sem atenção prolongada) e estímulos endógenos sucessivos (com atenção prolongada). Fonte: (COLOMBINI, 2014).

2.5.2 Sistema cognitivo

O sistema cognitivo do modelo proposto por Simões (2015) compreende blocos de memórias (semântica, episódica, procedimental e de trabalho), de desempenho, de individualidade (estados internos, esquema corporal), dentre outros módulos cognitivos. A memória de trabalho recebe informações provenientes do sistema atencional (mapa de saliências e memória sensorial) e provê informações para blocos de motivação, bloco de emoções, bloco de tomada de decisão, bloco de desempenho e bloco de outros processos cognitivos. As memórias semântica, episódica e procedimental trocam informações com a memória de trabalho num fluxo bidirecional (SIMÕES; COLOMBINI; RIBEIRO, 2016).

Além dos sistemas, um conjunto de processos realiza ações sobre os elementos do sistema cognitivo paralelamente, sincronizados com os módulos. O sistema opera sobre o paradigma principal de que qualquer função ou processo do sistema cognitivo pode requisitar informações de qualquer outro módulo e/ou alterar estados internos do agente em qualquer tempo, não sendo obrigatória, portanto, a passagem do fluxo de informação pelo módulo de tomada de decisão (SIMÕES; COLOMBINI; RIBEIRO, 2017).

2.5.3 Formalização

O modelo foi descrito formalmente em Simões (2015), conforme segue:

- Um **conjunto de ações** do agente, com um número arbitrário de elementos denominados ações;
- Um **conjunto de ações atencionais** do agente, com um número arbitrário de elementos denominados ações atencionais, que realimentam o sistema de seleção para percepção de acordo com os propósitos correntes do agente;

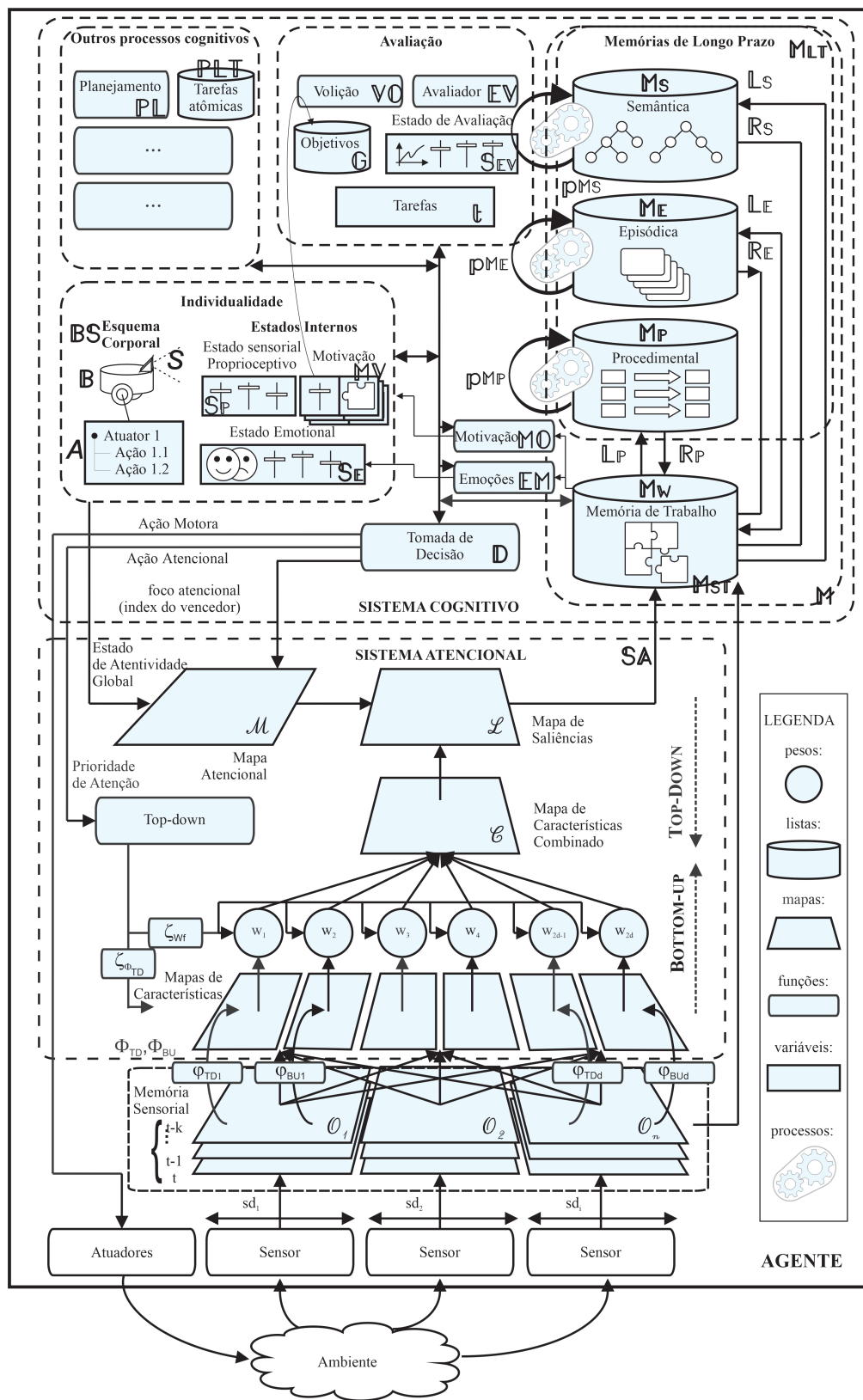


Figura 5 – Sistema cognitivo-atencional CONAIM. Fonte: Simões (2015).

- Um **conjunto de ações motoras** do agente, com um número arbitrário de elementos denominados ações motoras, que influenciam os atuadores do sistema;
- Uma **lista simples de objetivos** do agente, que contém um número arbitrário de elementos;
- Um **conjunto de tarefas** do agente, com um número arbitrário de elementos denominados tarefas, com uma tarefa específica denominada tarefa corrente do agente;
- Um **conjunto de estados** do agente, com um número arbitrário de elementos denominados estados;
- Um **conjunto de estados atencionais** do agente, com um número arbitrário de elementos, onde cada elemento, denotado estado atencional, é constituído de um mapa de saliências, conforme o proposto no modelo de [Colombini \(2014\)](#);
- Um **conjunto de estados de desempenho** do agente, com um número arbitrário de elementos, onde cada elemento, denotado estado de desempenho, é constituído de um número arbitrário de variáveis de desempenho do agente;
- Um **conjunto de motivações do agente**, com um número arbitrário de elementos denominados motivações;
- Um **super conjunto denominado memória**, com um conjunto denominado memória de curto prazo, um conjunto denominado memória de longo prazo, uma lista duplamente encadeada de elementos denominada memória de trabalho, uma lista duplamente encadeada de elementos denominada memória semântica, uma lista duplamente encadeada de elementos denominada memória episódica, uma lista duplamente encadeada de elementos denominada memória procedimental, um conjunto de qualificadores e um conjunto estado-ação;
- Uma **função de planejamento**, que mapeia um objetivo para um conjunto com um número arbitrário de tarefas;
- Uma **função de avaliação** do agente, que mapeia o estado corrente do agente, seus objetivos e tarefas;
- Uma **função de motivação** do agente, que mapeia o conteúdo da memória de trabalho para a motivação;
- Uma **função de tomada de decisão** do agente, que tipicamente coordena todos os sistemas cognitivos, listas e elementos escolhendo uma ação (que pode ser uma ação motora ou uma ação atencional) e um índice denotado índice do Vencedor do processo atencional ou foco da atenção.
- Um **esquema corporal do agente**, formado pela junção entre o conjunto de sensores do agente, o conjunto de atuadores do agente, e um conjunto de características descritivas da estrutura física do agente (peso, volume, centro de gravidade, formato externo etc.);
- Um **conjunto de processos assíncronos inconscientes**, responsáveis por realizar um mapeamento de elementos específicos da arquitetura para atualização de estados e/ou listas do sistema cognitivo;
- Aprendizado e recordação semânticos;
- Aprendizado e recordação episódicos;
- Aprendizado e recordação procedimentais;

2.5.4 Operação

A operação do modelo CONAIM é descrita em [Simões \(2015\)](#), conforme segue:

1. Iniciar novo ciclo atencional.

- a) Computar o modelo de [Colombini \(2014\)](#) até a geração do mapa de saliências;
- b) Computar o Vencedor do processo competitivo (*Winner*), isto é, o índice do sensor com a máxima saliência;

2. Iniciar novo ciclo cognitivo.

- a) Criar um novo elemento na memória de trabalho e inserir o índice do sensor vencedor, suas observações e o mapa de saliências;
- b) Computar o aprendizado semântico visando a inserção ou ampliação de conceitos;
- c) Computar o aprendizado episódico visando a inserção ou ampliação de episódios;
- d) Computar o aprendizado procedimental visando a inserção ou ampliação de procedimentos;
- e) Computar a recordação semântica, inserindo o conteúdo recordado na memória de trabalho. Pode ser realizada baseada em muitos aspectos distintos (forma, cor, cheiro, propósito, número de *recalls* etc.). Para cada aspecto, um algoritmo distinto de correspondência deve realizar uma busca completa nos elementos da memória;
- f) Computar a recordação episódica, inserindo o conteúdo recordado na memória de Trabalho. Pode ser realizada baseada em muitos aspectos distintos (tempo do evento, condições ambientais, objetos, ações realizadas etc.). Para cada aspecto, um algoritmo distinto de correspondência deve realizar uma busca completa nos elementos da memória;
- g) Computar a recordação procedimental, inserindo o conteúdo recordado na memória de trabalho. Pode ser realizada baseada em muitos aspectos distintos (propósitos, restrições, condições etc.). Para cada aspecto, um algoritmo distinto de correspondência deve realizar uma busca completa nos elementos da memória;
- h) Computar a função motivação. Funções de motivação podem ser realizadas com base em um grande número de teorias psicológicas. Tipicamente, a motivação é reduzida quando certa atividade é realizada por muito tempo, ou se ela é entendida como difícil ou se o agente tem atividades mais agradáveis a serem realizadas. Elas também dependem do agente como um indivíduo, de seu estado emocional e memórias ou de sua curiosidade com relação a um novo estímulo ([SIMOES; COLOMBINI; RIBEIRO, 2017](#));
- i) Se necessário, computar a função de planejamento para determinar as tarefas atômicas que precisam ser realizadas para cumprir os objetivos;
- j) Avaliando os objetivos, a tarefa corrente do agente, o estado geral do agente e as tarefas que precisariam ser executadas para o atendimento das tarefas, computar o estado de desempenho do agente;
- k) Computar a tomada de decisão, fazendo:
 - i. Considerando a lista de objetivos, a tarefa corrente do agente, o estado geral do agente e as tarefas que precisariam ser executadas para o atendimento dos objetivos, deliberar sobre a manutenção ou alteração da tarefa corrente;
 - ii. Inserir como item na memória de trabalho e aguardar a recordação da tarefa;
 - Se o procedimento para resolução da tarefa for conhecido, então executar a tarefa utilizando o procedimento descrito no conjunto de pares estados-ação;
 - Caso contrário, se a recordação retornar falha, avaliar se o agente deve realizar um procedimento de aprendizado da tarefa. Em caso afirmativo, criar um novo elemento e iniciar o aprendizado.

3. Iniciar novo ciclo atencional para $tempo=tempo+1$. (Reinício do processo)

Além da execução do modelo, três processos assíncronos inconscientes são executados em *background*:

- Um processo que varre permanentemente a memória semântica buscando derivar ou unificar conceitos;
- Um processo que varre permanentemente a memória episódica buscando derivar ou unificar episódios;
- Um processo que varre permanentemente a memória procedimental buscando derivar ou unificar procedimentos.

2.6 CST

CST O CST (*Cognitive Systems Toolkit*) (GUDWIN et al., 2013; PARAENSE; RAIZER; GUDWIN, 2016; PARAENSE et al., 2016) é um kit de ferramentas, baseado em Java, em repositório aberto (GUDWIN et al., 2019), para a construção de arquiteturas cognitivas. O CST permite o uso e integração de diversas tecnologias computacionais encontradas em outras arquiteturas, como CLARION e LIDA. O núcleo do CST consiste em um conjunto de conceitos base que podem ser utilizados de forma generalizada dentro de qualquer arquitetura cognitiva a ser construída (PARAENSE et al., 2016). Suas funções cognitivas são classes que podem ser combinadas em um conjunto paralelo de dispositivos de interação. As ferramentas do CST permitem a criação de sistemas multiagentes em execução assíncrona e paralela. Seu diagrama de classes pode ser visto na Figura 6.

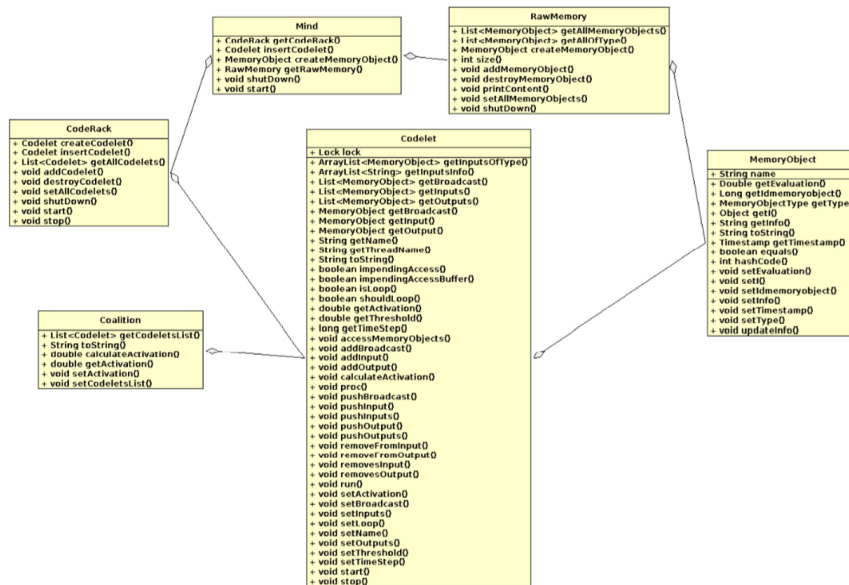


Figura 6 – Diagrama de classes do CST Toolkit. Fonte: (GUDWIN et al., 2013).

2.6.1 Codelets

Codelets A orientação do CST ocorre através de *codelets*, pequenos pedaços de código, implementados como funções assíncronas, que são executados paralelamente com tarefas simples e definidas (RAIZER; PARAENSE; GUDWIN, 2011; RAIZER; PARAENSE; GUDWIN, 2012; PARAENSE et al., 2016). Os *codelets* representam todas as principais funções cognitivas, ocorrendo constantemente e ciclicamente, e são responsáveis pelos comportamentos do sistema. Os *codelets* são semelhantes aos processos descritos na Teoria do Espaço de Trabalho Global (GWT) de Baars e aos aplicados na arquitetura LIDA (BAARS, 1998; PARAENSE; RAIZER; GUDWIN, 2016). Os *codelets* são organizados

em grupos, cada grupo implementa um modelo cognitivo de uma função cognitiva (PARAENSE et al., 2016). O kit de ferramentas CST fornece implementações padrão para alguns desses grupos.

Cada *codelet* no CST possui duas entradas, Local (Li) e Global (Gi). A entrada **Local (Li)** é a entrada de informações padrão sobre objetos de memória selecionados e normalmente é fixa, mas pode ser alterada através de mecanismos de aprendizado (GUDWIN et al., 2013; PARAENSE; RAIZER; GUDWIN, 2016). A entrada **Global (Gi)** é utilizada para obter informações do Espaço de Trabalho Global (ETG) e é atualizada a todo momento devido aos mecanismos de consciência (PARAENSE et al., 2016).

Os *codelets* no CST também possuem duas saídas, Padrão (O) e Nível de Ativação (A). A saída **Padrão (O)** é utilizada para alterar ou criar novas informações na memória (PARAENSE et al., 2016). A saída **Nível de ativação (A)** indica a relevância das informações obtidas na saída Padrão e é utilizada para selecionar as informações que serão utilizadas no ETG.

2.6.2 Objetos de memória

Um objeto de memória (*Memory Object* - MO) é um sinal ou representação interna utilizado, com outros MOs, pelos *codelets* para armazenar e acessar dados. A principal propriedade de um MO é sua informação (I), modelada como um objeto genérico em Java, e podendo ser representada em um valor unitário, em listas ou coleções (GUDWIN et al., 2017). Cada MO também possui um registro de tempo (T), que indica sua última atualização, e uma avaliação (E) (PARAENSE et al., 2016; RAIZER; PARAENSE; GUDWIN, 2011).

Algumas funções cognitivas exigem mais prioridade que outras e processos específicos devem ser executados antes de outros, por exemplo, o processo sensorial antes dos processos perceptivos. No caso do CST, muitos *codelets* podem escrever no mesmo MO ao mesmo tempo. Através do *Memory Container*, um MO aprimorado, ilustrado na Figura 7, vários *codelets* podem escrever ao mesmo tempo, com objetos separados para entrada e um mecanismo padrão para saída (GUDWIN et al., 2013). O *codelet* de seleção, implicitamente incorporado, seleciona o MO que contém o maior campo de avaliação (E).

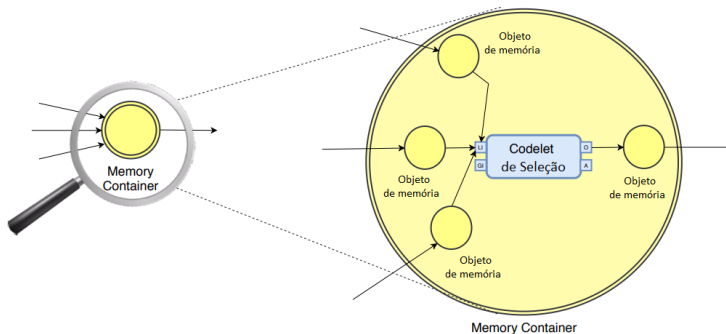


Figura 7 – *Memory Container* no CST Toolkit. Fonte: (BERTO, 2020) (adaptado).

2.6.3 Mind

A classe *Mind* do CST representa o núcleo do CST e é formada por dois contêineres: *Coderack* e o *Raw Memory* (PARAENSE; RAIZER; GUDWIN, 2016; PARAENSE et al., 2016). O *Coderack* armazena os *codelets* existentes no sistema e representa o espaço de trabalho global na Teoria de Espaço de Trabalho Global (*Global Workspace Theory* - GWT) e a *Raw Memory*, armazena os MOs. Os *codelets* interagem uns com os outros através dos MOs e podem gerar coalizões. Por exemplo, um MO pode ser a saída de um *codelet* e a entrada de outro.

As entradas principais dos *codelets* podem ser usadas em conjunto para implementar o mecanismo do Espaço de Trabalho Global (ETG) de consciência (PARAENSE; RAIZER; GUDWIN, 2016; PARAENSE et al., 2016). A entrada local é uma lista de MOs selecionados da *Raw Memory* e o *codelet* é capaz de obter as informações necessárias para realizar seu *proc()* de processamento. De acordo com a GWT, os *codelets* que estão no ETG são transmitidos para todos os *codelets* no sistema através da entrada global.

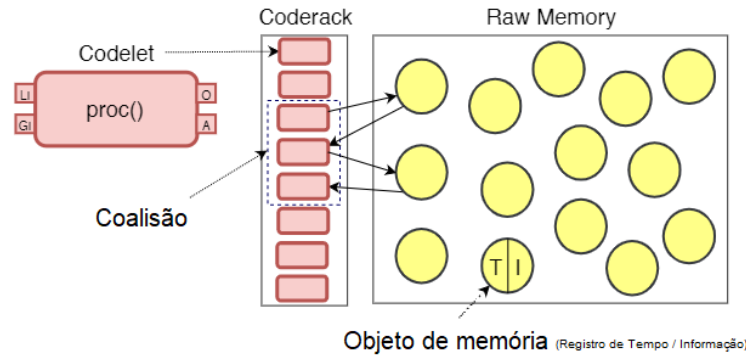


Figura 8 – Núcleo do CST Toolkit. Fonte: (PARAENSE et al., 2016) (adaptado).

2.6.4 Coderack

Diversos tipos de *codelets* integram o *Coderack* e são prescritos pela Arquitetura Cognitiva de Referência do CST (PARAENSE et al., 2016):

- **Codelets sensoriais:** responsáveis por obter informações dos sensores no ambiente e alimentar os MOs correspondentes;
- **Codelets perceptivos:** responsáveis por gerar percepções, ou representações de alto nível, para o ambiente. São alimentados pelos MOs na memória sensorial e alimentam MOs na memória perceptiva;
- **Codelets de atenção:** tipos especializados de *codelets* que funcionam como detectores de saliência para objetos ou situações no ambiente e que podem ser importantes para a definição de uma estratégia de ação ou comportamento. Estes *codelets* capturam informações de MOs na memória perceptiva e alimentam MOs na própria memória perceptiva ou na memória de trabalho;
- **Codelets de consciência:** Conjunto de *codelets* que implementa o espaço de trabalho global (ETG), de forma semelhante à LIDA. Porém, enquanto na LIDA os nós de uma coalisão são aqueles que disparam ao mesmo tempo, no CST os *codelets* assumidos em uma coalisão estão acoplados por meio de um MO comum;
- **Codelets Motores:** responsáveis por captar MOs da memória motora e transformá-los em atuações no ambiente. Normalmente, esses MOs motora são gerados por *codelets* comportamentais anteriores.

2.6.5 Raw Memory

Os MOs são utilizados para armazenar e acessar diferentes tipos de conhecimento e estão distribuídos entre diferentes tipos de memórias, que integram a *Raw Memory*. Os seguintes tipos de memórias são prescritos pela Arquitetura Cognitiva de Referência CST (PARAENSE et al., 2016):

- **Memória sensorial:** armazenamento bruto de informações provenientes de modalidades visuais e outras sensoriais em um período de tempo de 50 a 500 ms. Esta memória contém MOs que possuem representações diretas dos sensores do sistema. Esses MOs podem ser números simples ou estruturas de dados complexas, representando sensores escalares ou imagens n-dimensionais. Os MOs armazenados na memória sensorial são atualizados por *codelets* sensoriais. A memória também pode armazenar sequências temporais de dados de sensores, que podem ser usados por *codelets* perceptuais para criar percepções mais elaboradas;
- **Memória motora:** memória similar à memória sensorial, mas para os atuadores do sistema. memórias objetos na memória motora são valores utilizados como parâmetros dos atuadores a fim de atuar no ambiente;
- **Memória perceptiva:** memória de categorias, atributos, abstrações de alto nível e padrões que podem ser percebidos por um sistema perceptivo. Cada instância é uma representação de uma categoria. A memória perceptual é alimentada por *codelets* perceptuais, que coletam informações de MOs sensorial e fornecem abstrações de alto nível de dados sensoriais em termos de percepções. Outras categorias de *codelets* podem usar MOs perceptuais como fonte de informação;
- **Memória de trabalho:** tipo volátil de memória usada durante a percepção, raciocínio, planejamento e outras funções cognitivas. Sua capacidade no tempo e no espaço é curta, variando de 4 a 9 itens, e períodos até algumas dezenas de segundos. MOs representam mapas do ambiente, ou representações mais elaboradas, onde vários locais estão conectados formando uma rede inteira. Esses MOs podem ser usados por códigos de planejamento para derivar um plano de ações;
- **Memória procedimental:** memória das ações e comportamentos de um sistema. É uma memória não-declarativa que se refere a um tipo de informação de como executar determinadas ações, geralmente consistindo em um registro de possíveis habilidades motoras e comportamentais;
- **Espaço de trabalho global (ETG):** o Espaço de trabalho global é uma coleção de referências a outros MOs armazenados nas diferentes memórias descritas anteriormente e é constantemente alterado pelo mecanismo de consciência implementado pelos *codelets* da consciência.

2.7 CONAIM+CST *bottom-up attentional*

Um dos possíveis cenários para a implementação do CONAIM com as ferramentas do CST Toolkit foi investigado pelo grupo de trabalho em [Araújo e Gudwin \(2018\)](#), [Regattieri e Colombini \(2018\)](#) e [Berto \(2020\)](#). A classe *Mind* do CST é utilizada como sistema cognitivo e *codelets* e objetos de memória (MOs) são utilizados para a geração de elementos *bottom-up attentional* do sistema atencional, como mapa atencional, mapa de saliências, mapa de características, inibição de retorno (IOR) e a eleição de características vencedoras através de um algoritmo WTA. Os trabalhos [Regattieri e Colombini \(2018\)](#) e [Berto \(2020\)](#) também estudam a utilização destes elementos com uma classe de aprendizado que implementa o algoritmo de RL Q-Learning. O esquema da Figura 10 é apresentado por [Berto \(2020\)](#) para representar a implementação com MOs e *codelets* do CST do CONAIM *bottom-up attentional*. O diagrama minimiza a multiplicidade de relacionamentos, atribuindo cardinalidades de 1 ou n a eles.

2.7.1 Percepção e Atenção

Os *codelets* *GetSensor* e *SensorBuffer* representam interfaces mínimas para mensurar informações sensoriais em observações das dimensões dos sensores ($\mathcal{O}$), representadas por *Sensor* MOs e *SensorBuffer* MOs, respectivamente ([BERTO](#),

2020, 2020). Os *SensorBuffer* MOs integram *codelets FeatureMaps*, que geram mapas de características ($\mathcal{F}$), representados por *FeatMap* MOs, de acordo com os parâmetros de observação. Estes mapas são combinados através de uma soma ponderada pelo *codelet CombinedFeatureMap* em um mapa de características combinado ($\mathcal{C}$), representado pelo *CombFeatMap* MO.

Por meio do *codelet SaliencyMapCodelet*, o Mapa Atencional ($\mathcal{M}$), representado pelo *AttMap* MO e inicialmente criado com valores unitários para simular um ambiente sem vencedores, é multiplicado por ($\mathcal{C}$) para a formação de saliências das características mais interessantes no ambiente em um mapa de saliências ($\mathcal{L}$), representado pelo *SalMap* MO (COLOMBINI; SIMÕES; RIBEIRO, 2017). Estas saliências influenciam o *codelet* de tomada de decisões (*DecisionMaker*), que define a característica vencedora desse instante utilizando a abordagem *Winner Takes All* (WTA). Após a seleção, o *codelet* de tomada de decisões também adapta $\mathcal{M}$, através da supressão da atividade no local selecionado pelo mecanismo de IOR, e reiniciando o ciclo atencional.

2.7.2 Aprendizado Procedimental

Com as informações sobre os níveis de saliência e as características vencedoras do ciclo atencional, pode-se alimentar o *codelet* de aprendizado (*Learner Codelet*) para o agente aprender quais ações são mais aplicáveis a cada etapa (REGATTIERI; COLOMBINI, 2018; BERTO, 2020, 2020). A classe *Learner* é implementada com a classe QLearning do CST (GUDWIN et al., 2013; GUDWIN et al., 2019). A classe QLearning requer a especificação do Processo de Decisão de Markov (MDP) associado, em termos de ações, estados e recompensas. Os estados possíveis estão diretamente vinculados ao mapa de saliências $\mathcal{L}_t \Rightarrow S_t$, de t dimensões. Após a geração do Mapa de Saliências, cada elemento $l_i \in \mathcal{L}_t$ é normalizado em uma escala de de 0 a 1:

$$l_i = \frac{l_i - \min(\mathcal{L}_t)}{\max(\mathcal{L}_t) - \min(\mathcal{L}_t)} \quad (7)$$

Após a normalização, cada elemento l_i do mapa de saliências $\mathcal{L}_t$ é discretizado em τ possíveis faixas (a cada $\lambda = 1/\tau$), com $i \in [0, t-1]$. Cada uma destas faixas possui um multiplicador correspondente $n_i \in N_t$:

- $l_i \leq \lambda \Rightarrow n_i = 0$;
- $\lambda < l_i \leq 2\lambda \Rightarrow n_i = 1$;
- $2\lambda < l_i \leq 3\lambda \Rightarrow n_i = 2$;
- ...
- $(1-\lambda) < l_i \leq 1 \Rightarrow n_i = \tau - 1$;

Cada um dos multiplicadores n_i é multiplicado por τ^{i-1} e resulta em $d_i \in D_t$. O estado S_t é representado pela somatória dos elementos de D_t . O processo é ilustrado na Figura 9. Os estados (S_t) e as possíveis ações do agente (A_t) são utilizados no *Learner Codelet* para a criação da tabela de aprendizado QTable do algoritmo QLearning e, posteriormente, com as recompensas obtidas a cada ciclo, para a atualização da QTable.

A QTable é inicializada com um valor de avaliação $Q_{old}(s_i, a_i)$ nulo para todos os estados com todas as ações possíveis. A cada iteração do *Learner Codelet*, o estado e a ação do instante anterior s_{t-1} e a_{t-1} são adquiridos, assim como a recompensa r_t , referente à recompensa obtida com a aplicação da ação a_{t-1} no estado e_{t-1} . Estes valores são utilizados com $\alpha = 0,5$ e $\gamma = 0,9$ na Equação 1 para o cálculo do novo valor de avaliação $Q_{new}(s_{t-1}, a_{t-1})$ (BERTO, 2020).

Após a atualização, a próxima ação a_t é selecionada com s_t como estado. Esta ação possui uma probabilidade de 10% de ser a ação mais bem avaliada de acordo com a QTable. Os 90% restantes são divididos entre 85,5% da ação ser

a ação anterior a_{t-1} e 4,5% da ação ser uma ação randômica $\in A$. Cada ação selecionada possui um *Behavioral Codelet* correspondente, que avalia a possível utilização da ação com base em seu comportamento e no instante atual. As ações e suas avaliações são enviadas para o *Memory Container*, onde são armazenadas na memória motora e participam do processo de seleção para definir qual ação deve ser executada pelo *Motor Codelet* (BERTO, 2020).

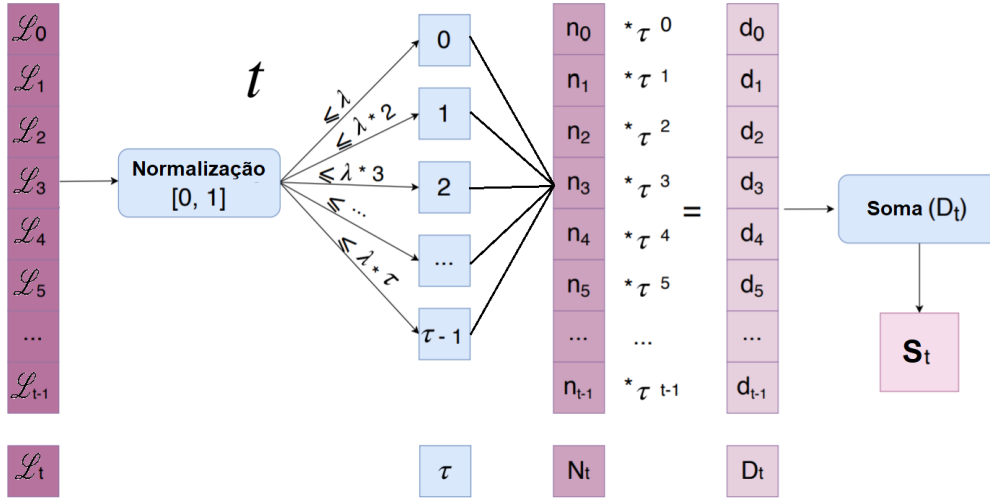


Figura 9 – Esquema ilustrando o processo de discretização do Mapa de Saliências. Fonte: (BERTO, 2020) (adaptado).

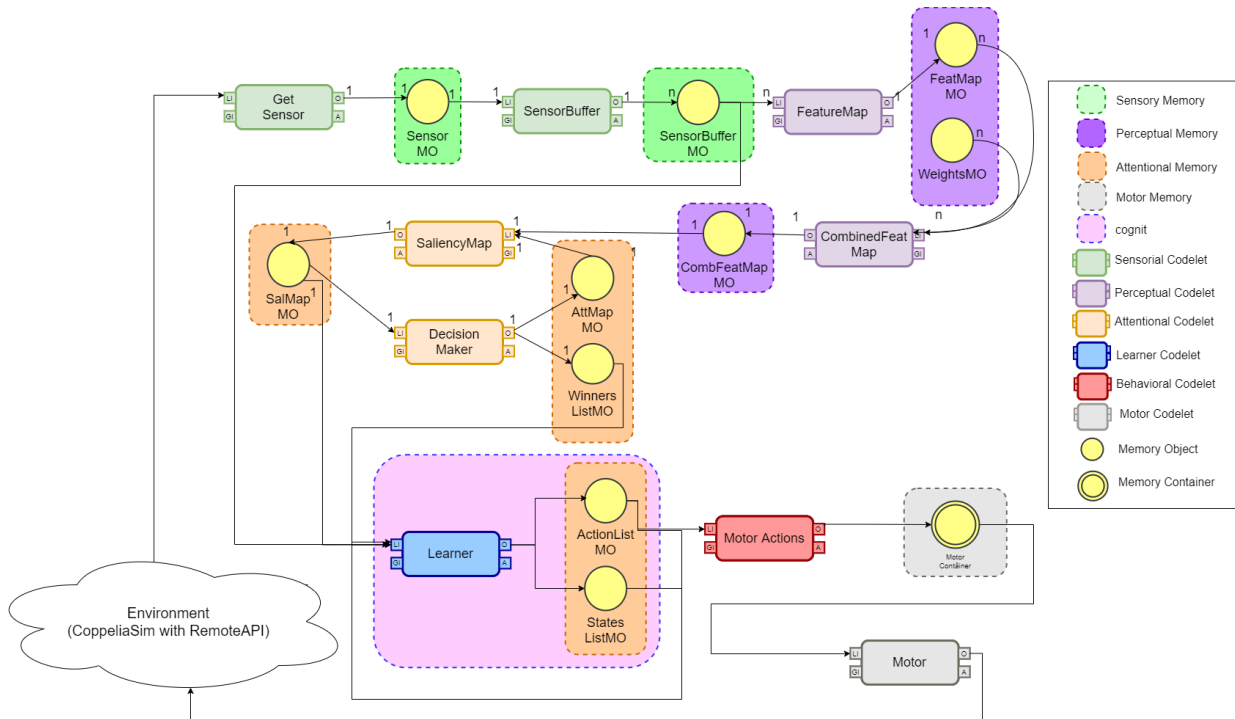


Figura 10 – Esquema de aprendizado procedimental com CONAIM *bottom-up attentional* implementado com CST. Fonte: (BERTO, 2020) (adaptado).

2.8 Comparação entre Arquiteturas

A partir do compilado de arquiteturas apresentado por [Kotseruba e Tsotsos \(2020a\)](#), nota-se um interesse particular na implementação de arquiteturas e estruturas simbólicas desde meados da década de 80 até o início da década de 90. No entanto, após os anos 2000, a maioria das arquiteturas e estruturas recém-desenvolvidas é híbrida. Modelos emergentes, muitos dos quais com inspiração biológica, são distribuídos de maneira mais uniforme, mas permanecem em um grupo relativamente pequeno ([KOTSERUBA; TSOTSOS, 2020a](#)). Dentre as arquiteturas cognitivas híbridas mais populares, que possuem uma longa tradição de desenvolvimento em seus grupos de pesquisa e que fornecem documentação e estruturas computacionais gratuitamente, podem-se citar as arquiteturas CLARION (*Connectionist Learning with Adaptive Rule Induction On-line*), SOAR (*State, Operator and Result*) e LIDA (*Learning Intelligent Distribution Agent*).

Na Tabela 1 são apresentadas comparações entre os principais aspectos das arquiteturas SOAR, CLARION, LIDA e na implementação da arquitetura CONAIM *bottom-up attentional* utilizando as ferramentas cognitivas do CST Toolkit.

Comparação entre arquiteturas cognitivas.				
Parâmetros	CONAIM+CST <i>bottom-up attentional</i>	SOAR	CLARION	LIDA
Principais referências	(ARAÚJO; GUDWIN, 2018; REGATTIERI; COLOMBINI, 2018; BERTO, 2020)	(LAIRD, 2008; LAIRD et al., 2017; LAIRD, 2012; MININGER; LAIRD, 2016)	(SUN, 2004; SUN; FRANKLIN, 2016; SUN; WILSON; LYNCH, 2019)	(FRANKLIN, 2012; FRANKLIN; FAGHIHI; MCCALL, 2012; SNAIDER et al., 2012)
Representação do conhecimento e percepção	Informações sensoriais armazenadas em MOs. <i>Codelets</i> atencionais geram de mapas de características, um mapa atencional e um mapa de saliências, que definem os índices de características vencedoras em cada ciclo atencional. Conhecimento procedimental representado por <i>codelets</i> de comportamento que promovem a seleção de ações.	Informações sensoriais armazenadas em elementos de memória de trabalho (EMT). Conhecimento procedimental modelado como regras. Conhecimento semântico modelado como uma rede estrutural de EMTs e conhecimento episódico modelado como episódios dessas estruturas.	Informações sensoriais armazenadas em pares de dimensão-valor que representam o tipo da informação e seu valor correspondente. O conhecimento procedimental é armazenado como regras, redes neurais e <i>chunks</i> no subsistema centrado em ações (SCA). O conhecimento declarativo é armazenado como <i>chunks</i> e elementos de baixo nível no subsistema não-centrado em ações (SNCA).	Informações sensoriais ativam detectores de características. O conhecimento perceptual é gerado por <i>codelets</i> através da avaliação destas características e de seu armazenamento em nós da memória perceptual associativa (MPA). O conhecimento episódico é representado por estruturas relacionais destes nós na memória episódica transiente (MET). O conhecimento procedimental é representado por esquemas de ações.
Componentes principais	Ciclo cognitivo-atencional realiza a seleção de ações, otimizada através do <i>codelet</i> de aprendizado. Memórias armazenadas na <i>Raw Memory</i> e <i>Codelets</i> armazenados no <i>Coderack</i>	Memória de trabalho é codificada como uma estrutura de EMTs.	Subsistema centrado em ações (SCA) e subsistema não-centrado em ações (SNCA).	Ciclo cognitivo realiza a seleção de ações e compreende memória de trabalho (espaço de trabalho), memória sensorial, memória perceptual associativa, memória declarativa, memória procedimental, memória sensório-motora e espaço de trabalho global (ETG).
Memória de trabalho	MOs ativos na <i>Raw Memory</i>	Organizada como uma estrutura associada de EMTs.	Integra o subsistema centrado em ações (SCA), seu nível explícito é formado por <i>chunks</i> e seu nível implícito é formado por elementos relacionados a estes <i>chunks</i> .	Espaço de trabalho pré-consciente, incluindo modelo atual situacional (MAS) e pilha de conteúdo consciente.
Memória declarativa	Associações entre MOs na <i>Raw Memory</i>	Dividida entre memória semântica (associações entre EMPs) e memória episódica (episódios destas associações).	Subsistema não-centrado em ações (SNCA), seu nível implícito formado por redes neurais que representam características do ambiente e seu nível explícito formado por <i>chunks</i> destas características.	Associações de memória esparsamente distribuída. Dividida entre memória semântica e memória episódica.
Memória procedimental	<i>Codelets</i> de comportamento no <i>Coderack</i> que promovem a seleção de ações	Regras de ação.	Subsistema centrado em ações (SCA).	Esquemas de ação.
Sistema de recompensas	Aprendizado <i>Q-Learning</i> no <i>codelet Learner</i>	Aprendizado por reforço nos operadores de acordo com níveis de ativação.	Subsistema meta-cognitivo (SMC) determina recompensas de acordo com o subsistema motivacional (SM).	Aprendizado por reforço nos níveis base de ativação para memória perceptual associativa (MPA), memória episódica e memória procedimental.
Seleção de ações	Ações definidas pelo sistema comportamental e aplicadas de acordo com as avaliações de ações no algoritmo <i>Q-Learning</i>	Ações associadas a operadores e aplicadas de acordo com suas preferências e o estado atual.	Ações propostas pelas camadas explícita e implícita do subsistema centrado em ações (SCA) e selecionadas de acordo com um método probabilístico	Ações propostas pela memória procedimental e apenas uma ação é selecionada pelo subsistema de seleção de ações.

Tabela 1 – Comparação entre arquiteturas cognitivas. Adaptado de (SAMSONOVICH, 2010a).

3 ROBÓTICA DO DESENVOLVIMENTO

"Houve um tempo em que o homem enfrentou o universo sozinho e sem amigos. Agora ele tem criaturas para ajudá-lo. Criaturas mais fortes que ele próprio, mais fiéis, mais úteis e totalmente devotadas a ele. A humanidade não está mais sozinha."

(Isaac Asimov)

Eu, robô.

A robótica do desenvolvimento (*developmental robotics*), robótica epigenética ou *devRobotics*, é um campo multidisciplinar da robótica que utiliza de teorias dos desenvolvimentos humano e animal elaboradas em áreas como psicologia do desenvolvimento, neurociência, biologia e linguística, para formalizá-las e implementá-las em robôs, explorando extensões ou variantes delas para que a arquitetura de controle possa se adaptar ao seu ambiente (LUNGARELLA; SANDINI; PFEIFER, 2003; STOYTCHEV, 2009; VERNON; HOFSTEN; FADIGA, 2016). A área surgiu devido à necessidade de robôs aumentarem a escala de tarefas que requerem níveis próximos de inteligência humana em ambientes complexos e imprevisíveis, envolvendo adaptação e evolução (STOYTCHEV, 2009). Os modelos e experimentos da área são inspirados pelos princípios e mecanismos de desenvolvimento observados no começo da vida, envolvendo robôs desempenhando as mesmas habilidades cognitivas de crianças, como nos experimentos propostos por (PIAGET, 1952). De acordo com Jean Piaget, as habilidades cognitivas humanas emergem incrementalmente (CANGELOSI; SCHLESINGER, 2014; LUNGARELLA; SANDINI; PFEIFER, 2003). Esta seção busca apresentar a Robótica do Desenvolvimento, abordando as características encontradas nos projetos da área e apresentando a Teoria de Piaget (1952), a Analogia de Berto (2020) (vide seção 3.6) e suas propostas de experimentos sensório-motores para a avaliação de arquiteturas cognitivas no âmbito da robótica do desenvolvimento.

3.1 A Teoria de Piaget

A área da Robótica do Desenvolvimento possui inspirações derivadas das teorias de Jean Piaget (1896 – 1980), um psicólogo suíço que postulou a teoria da **epistemologia genética** com o objetivo de explicar a continuidade entre processos biológicos e cognitivos humanos e como comportamentos mais simples derivam experiências mais complexas (PIAGET, 1952; PIAGET, 1971).

Um conceito relevante na teoria de Piaget são os **esquemas**, redes de estruturas mentais que ajudam alguém a relembrar conceitos específicos e a entender o ambiente (ARMSTRONG et al., 2014). Segundo Piaget (1999), quando processos mentais simples se tornam mais sofisticados, novos esquemas são desenvolvidos e o comportamento também se torna mais complexo e adequado ao ambiente.

No desenvolvimento mental de Piaget (1952), existem elementos variáveis e elementos invariáveis. Os elementos variáveis são as construções mentais que o sujeito efetua ao longo de seu desenvolvimento. Os elementos invariáveis seguem duas tendências básicas: a organização e a adaptação. A **organização** é a tendência de um indivíduo em organizar seus processos de pensamento em estruturas psicológicas (PIAGET, 1952). De acordo com Woolfolk, Winne e Perry (2010), estruturas simples são combinadas entre si para se tornarem mais eficazes e sofisticadas. A **adaptação** ou **aprendizado** é a tendência em ajustar processos mentais ao ambiente através da alteração de estruturas cognitivas (COOK; COOK, 2005). O processo de adaptação envolve a equilibração entre os processos de assimilação e acomodação (LEFRANÇOIS, 2008).

Assimilação e acomodação são processos descritos por Piaget (1999) como inseparáveis, complementares e simultâneos. **Assimilação** é a criação de novos esquemas seguindo o mesmo ciclo ou sequência dos esquemas já existentes

para interpretar a experiência e tomar decisões (PIAGET, 1952; PIAGET, 1999). Estes novos esquemas são induzidos à repetição pela assimilação. Como destacam Woolfolk, Winne e Perry (2010), somente ocorrerá assimilação se o esquema não conflitar com algo já estabelecido como parte integrante do que o sujeito já conhece. **Acomodação** é o processo complementar que envolve a alteração de esquemas existentes como resultado de novas informações adquiridas por meio da assimilação (PIAGET, 1952). Neste caso, o sujeito tentará resolver uma determinada situação modificando as estruturas antigas e buscando novas maneiras de agir para dominar novos cenários (PIAGET, 1999; WOOLFOLK; WINNE; PERRY, 2010).

Acomodação

Por exemplo, uma criança que já conhece um gato e possui esquemas sobre as características do animal. Quando essa criança vir pela primeira vez um gambá que é parecido com um gato, ela tentará relacionar a situação baseada nas estruturas existentes (do gato). Ao ser corrigida ou descobrir que o gambá é um animal diferente do gato, ela irá, por acomodação, modificar suas estruturas mentais para correlacionar e acomodar as novas informações a respeito do gato e, por assimilação, construir novos esquemas a respeito do animal gambá.

Equilibração é a competição entre assimilação e acomodação, representa o esforço para atingir um equilíbrio estável (PIAGET, 1952). Quando os esquemas assimilados não são suficientes para desempenhar novas experiências ou não proporcionam o total entendimento do mundo, a criança fica em estado de desequilíbrio. A equilibração então promove a disputa entre assimilação e acomodação com novas experiências para estes esquemas. A criança restaura o equilíbrio ao adaptar e organizar os novos esquemas que integram a experiência. Durante a vida toda, a busca pelo equilíbrio é a força motivadora do crescimento cognitivo (PAPALIA; FELDMAN, 2013).

Equilibração

A assimilação pode originar **reações circulares**, repetições de ciclos adquiridos ou em processo de aquisição (PIAGET, 1952). A reação circular resulta da assimilação de um resultado interessante, antes desconhecido pelo sujeito, que foi produzido através de uma redescoberta ou repetição de ação. As reações circulares são processos pelos quais o indivíduo aprende a reproduzir ocorrências desejadas originalmente descobertas ao acaso e que provocaram um resultado satisfatório, podendo ser primárias, secundárias ou terciárias (PIAGET, 1952).

Reações
Circulares

- **Reações Circulares Primárias:** condutas derivadas de reflexos, atividades do próprio corpo que formam novos esquemas através da coordenação dos sentidos (PIAGET, 1952). Como exemplo desta reação, Wadsworth (1997) cita o ato do bebê chupar o dedo acidentalmente e depois repetir a ação intencionalmente. Esta repetição é impulsionada por uma reação de prazer no bebê;
- **Reações Circulares Secundárias:** são derivadas de comportamentos propositais que direcionam o interesse para resultados exteriores e não mais para o corpo do bebê (PIAGET, 1952). Como exemplos desta reação, Papalia e Feldman (2013) citam as ações do bebê de apertar repetidas vezes uma mesma tecla de um piano de brinquedo ou de pegar um brinquedo e colocá-lo na boca;
- **Reações Circulares Terciárias:** são um prolongamento das reações circulares secundárias (PIAGET, 1952). Esforço do sujeito em apreender novidades em situações já conhecidas e buscar novas experiências por tentativa e erro. Como exemplo desta reação, Papalia e Feldman (2013) citam a ação do bebê de apertar diferentes teclas de um piano de brinquedo, percebendo as diferenças entre os sons.

Reações
Circulares
PrimáriasReações
Circulares
SecundáriasReações
Circulares
Terciárias

Na teoria de Piaget, o desenvolvimento do aprendizado depende de herança genética e da interação do indivíduo com o ambiente, com outros indivíduos e com objetos (PIAGET, 1952). Cada interação participa do desenvolvimento do conhecimento e das estruturas de inteligência (ARMSTRONG et al., 2014). Em seus estudos e análises do perfil cognitivo de crianças, Piaget dividiu **estágios** de desenvolvimento em períodos definidos, onde eram verificadas as capacidades de acompanhamento e manipulação de objetos e a busca por fontes de luz ou sonoras (PIAGET, 1952; PIAGET, 1971; WOOD; SMITH; GROSSNIKLAUS, 2011).

1. **Estágio sensório-motor:** Nascimento até o 2º ano de idade. Neste estágio, o bebê coordena as sensações

Estágio
sensório-motor

vivenciadas com comportamentos motores simples, percebe o mundo e atua nele (PIAGET, 1999). Esta etapa envolve atividades motoras sem a utilização de símbolos (PIAGET, 1952; PIAGET, 1971; PIAGET, 1999). As funções mentais são limitadas ao exercício de reflexos inatos, tendências instintivas, com seu universo circundando os limites de seus movimentos, como sucção e movimentar dos olhos. Progressivamente, a criança aperfeiçoa tais reflexos, adquire novas habilidades e se concebe dentro de uma realidade (PAPALIA; FELDMAN, 2013; WOOD; SMITH; GROSSNIKLAUS, 2011). O estágio sensório-motor é apresentado por Piaget (1952) em seis subestágios. As características de cada subestágio do estágio sensório-motor são apresentadas na Tabela 2;

Estágio
pré-operatório

2. Estágio pré-operatório: 2º ao 7º ano de idade. Durante este estágio, as crianças começam a desenvolver a linguagem, memórias e imaginação, podendo entender e expressar relações entre o passado e o futuro (PIAGET, 1971). A inteligência se destaca como sendo egocêntrica e intuitiva (PIAGET, 1999);

Estágio
operatório
concreto

3. Estágio operatório concreto: 7º ao 11º ano de idade. O desenvolvimento intelectual nesta fase é demonstrado através do uso de manipulação lógica e sistemática de símbolos relacionados a objetos concretos (PIAGET, 1971). O pensamento se torna menos egocêntrico com maior conscientização de eventos externos e envolve referências concretas (PIAGET, 1999);

Estágio
operatório
formal

4. Estágio operatório formal: Adolescência e idade adulta. Adolescentes e adultos utilizam de símbolos para representar o resumo de determinados conceitos (PIAGET, 1971). Adolescentes podem pensar em múltiplas variáveis sistematicamente, podendo formular hipóteses e abstrair conceitos já existentes (PIAGET, 1999).

3.2 Características da Robótica do Desenvolvimento

A robótica de desenvolvimento emergiu como reação à incapacidade das arquiteturas tradicionais de robôs em executarem tarefas que requerem alto nível de inteligência (STOYTCHIEV, 2009). Um dos principais problemas destas arquiteturas está relacionado ao aumento na complexidade da programação e engenharia de *software* que decorre do aumento da complexidade das tarefas que o robô deve executar. Conforme destaca Stoytchev (2009), a pré-programação não pode ser a solução para o problema de escalabilidade já que os ambientes que estes robôs operam são complexos e imprevisíveis.

A busca por novas metodologias orientadas para o avanço da robótica e para a construção de sistemas robóticos mais autônomos, adaptáveis e sociáveis promoveu o crescimento das relações e inspirações entre a psicologia do desenvolvimento e a robótica (ASADA et al., 2001). Com esta metodologia interdisciplinar, os robôs podem ser empregados como ferramentas de pesquisa e avaliação no desenvolvimento de modelos cognitivos para a psicologia, permitindo a predição de resultados e comparações quantitativas e qualitativas (BROOKS, 1997; CANGELOSI; SCHLESINGER, 2014). A incorporação de modelos baseados nestas teorias em robôs também permite a comparação do desempenho de agentes artificiais com a realidade, a validação das teorias do desenvolvimento humano e animal e possíveis novas hipóteses para a formação e o funcionamento da mente (CANGELOSI; SCHLESINGER, 2014; PFEIFER; SCHEIER, 1994).

A área da robótica do desenvolvimento visa projetar habilidades cognitivas e sensório-motoras, inspiradas da psicologia infantil, em robôs e investigar seu desenvolvimento incremental e a aquisição de novas habilidades cognitivas através de mecanismos motivacionais, atencionais, de interação e de exploração do conhecimento sensório-motor e do ambiente (LUNGARELLA; SANDINI; PFEIFER, 2003; CANGELOSI; SCHLESINGER, 2014). A área está relacionada, mas difere, da robótica evolutiva. Enquanto a robótica evolutiva utiliza populações que evoluem ao longo do tempo, *devRobotics* busca compreender como a organização do sistema de controle de um único robô pode se desenvolver através da experiência ao longo do tempo.

Conforme descreve Cangelosi e Schlesinger (2014), os experimentos da área podem ser categorizados com base na relação abordada entre a psicologia do desenvolvimento e a robótica. O primeiro caso envolve a modelagem direta

Estágio Sensorio-Motor.			
Período	Características	Causalidade (relação entre causa e efeito), Conceitos e Objetos	Exemplos de Comportamento
1º subestágio: Uso de reflexos. Nascimento até o 1º mês de vida (PIAGET, 1952).	Progressivamente, os recém-nascidos demonstram controle sobre seus reflexos inatos e iniciam comportamentos padrão mesmo quando estímulos normais não estão presentes (PIAGET, 1999). Estes reflexos, presentes desde o nascimento, são o início da assimilação mental. Os reflexos tornam-se mais complexos por integração de hábitos e percepções e proporcionam a conexão inicial entre os bebês e seus ambientes (PIAGET, 1999).	Os recém-nascidos não possuem noção de causalidade ou a noção de conceitos de objetos, sendo incapazes de diferenciar seus corpos do ambiente externo (WADSWORTH, 1997).	Ação de um bebê de sucção do peito da mãe quando este está em sua boca e a prensão de suas mãos quando a palma da mão é acariciada (PAPALIA; FELDMAN, 2013).
2º subestágio: Reações circulares primárias. 1º ao 4º mês de vida (PIAGET, 1952).	Surgem as reações circulares primárias, sensações corporais agradáveis que inicialmente foram obtidas ao acaso e repetidas propositalmente (PIAGET, 1952; PIAGET, 1999). Estas reações propiciam o desenvolvimento da capacidade de diferenciar objetos e de coordenar os membros em ações diferentes (WOOD; SMITH; GROSSNIKLAUS, 2011). O objeto de manipulação é o próprio corpo do bebê (COLE; COLE, 2003).	Os bebês são egocêntricos e se colocam como centro de causas e efeitos (WADSWORTH, 1997). Os bebês já conseguem acompanhar objetos em movimento e virar a cabeça em direção a ruídos (WADSWORTH, 1997).	A criança suga o polegar inicialmente em um movimento aleatório, depois repetindo essa ação devido à satisfação gerada (CHEC; MARTIN, 2002; PAPALIA; FELDMAN, 2013).
3º subestágio: Reações circulares secundárias. 4º ao 8º mês de vida (PIAGET, 1952).	Através da coordenação de informações visuais e motoras, o conhecimento sobre o meio é desenvolvido. Os bebês aprendem a repetir estímulos de prazer através da repetição de uma mesma ação ou por tentativa e erro e dirigem sua atenção para objetos e para os resultados de suas ações (CHEC; MARTIN, 2002; PIAGET, 1999). As ações são intencionais, inicialmente não orientadas para uma meta (PAPALIA; FELDMAN, 2013).	Os bebês se interessam pelo ambiente, entendem que os objetos são mais do que extensões de suas próprias ações e antecipam as posições espaciais de objetos em movimento (PIAGET, 1952; WADSWORTH, 1997).	Ato do bebê empurrar objetos para vê-los caindo no chão (PIAGET, 1999) ou de descobrir um brinquedo novo e o utilizar para brincar (WADSWORTH, 1997).
4º Subestágio: Coordenação de esquemas secundários. 8º mês ao 1º ano de vida (PIAGET, 1952).	Os bebês conduzem ações voltadas ao meio externo, desenvolvidas com objetivo e com intencionalidade (PIAGET, 1952; PIAGET, 1971). Os bebês começam a fazer generalizações a partir de experiências passadas para resolver novos problemas e conseguem combinar e coordenar esquemas elementares para atingir seus objetivos, podendo antecipar acontecimentos (COLE; COLE, 2003).	Os bebês entendem que objetos podem causar efeitos no ambiente e aprendem os conceitos de constância de forma e tamanho (WADSWORTH, 1997). No caso da permanência de objetos, se um objeto for demonstrado para um bebê e depois ser escondido, o bebê procurará primeiro no lugar onde o objeto foi demonstrado pela primeira vez, mesmo se depois ele o viu ser removido para outro lugar (PIAGET, 1952).	Ação do bebê de apertar repetidas vezes uma mesma tecla de um piano de brinquedo (PAPALIA; FELDMAN, 2013).
5º subestágio: Reações circulares terciárias. 1º ano ao 18º mês de vida (PIAGET, 1952).	Novos comportamentos são realizados e ações são variadas para obter resultados semelhantes (reações circulares terciárias) (WADSWORTH, 1997). As crianças demonstram originalidade na resolução de problemas e, por tentativa e erro, experimentam comportamentos até encontrar a melhor maneira de atingir uma meta (COLE; COLE, 2003). A criança começa a representar mentalmente uma realidade através de imagens mentais e símbolos, marcando o início do desenvolvimento do pensamento simbólico (CHEC; MARTIN, 2002).	Para a permanência de objetos, a criança procura o objeto no último lugar em que o viu escondido. Entretanto, a criança não procurará em um lugar onde não viu o objeto escondido (PIAGET, 1952).	Ação do bebê de apertar diferentes teclas de um piano de brinquedo, percebendo as diferenças entre os sons (PAPALIA; FELDMAN, 2013).
6º Subestágio: Combinações mentais. 18º mês ao 2º ano de vida (PIAGET, 1952).	Transição para o estágio pré-operacional. A capacidade de representação simbólica para armazenar e recordar objetos e ações na memória é incentivada, assim como a capacidade de pensar em ações antes de realizá-las (COLE; COLE, 2003; WOOD; SMITH; GROSSNIKLAUS, 2011). As crianças experimentam soluções em suas representações mentais (COLE; COLE, 2003; CHEC; MARTIN, 2002; WOOD; SMITH; GROSSNIKLAUS, 2011).	Para a permanência de objetos, a criança procurará um objeto que sumiu mesmo se não o viu escondido, demonstrando dominância na representação dos objetos ausentes e de seus deslocamentos (PIAGET, 1952).	Ação do bebê de brincar com uma caixa de encaixes, procurando o encaixe correspondente para cada formato (PAPALIA; FELDMAN, 2013).

Tabela 2 – Características do Estágio Sensorio-Motor de Piaget

de estudos da psicologia do desenvolvimento, em que o robô é diretamente construído para replicar experimentos psicológicos específicos. O segundo caso, foco do presente projeto, envolve a utilização de algoritmos inspirados em mecanismos cognitivos biológicos e sua análise em ambientes voltados ao aprendizado e desenvolvimento. Nestes últimos sistemas, o cérebro é formulado como um sistema intrinsecamente motivado e direcionado a objetivos que estimulam a formação de mecanismos sensorio-motores, sociais e de desenvolvimento.

3.3 Princípios do desenvolvimento

Os projetos da robótica do desenvolvimento assumem que a inteligência em sistemas naturais e artificiais segue princípios básicos da pesquisa no campo da psicologia do desenvolvimento (SUTTON; BARTO, 1998; STOYTCHEV, 2009; LUNGARELLA; SANDINI; PFEIFER, 2003; CANGELOSI; SCHLESINGER, 2014):

Corporificação

- **Corporificação:** sugere que um corpo, dotado de mente, está reciprocamente acoplado ao ambiente e fornece o substrato para exploração de hipóteses subjetivistas (BROOKS; STEIN, 1994; LUNGARELLA; SANDINI; PFEIFER, 2003). Processos cognitivos surgem deste corpo como capacidades perceptivas e motoras e promovem interação e movimento no ambiente (BEER, 2003; LUNGARELLA; SANDINI; PFEIFER, 2003; STOYTCHEV, 2009). Uma interpretação proposta por Stoytchev (2009) trata o corpo como a parte mais consistente, previsível e verificável do ambiente e, em qualquer trajetória de desenvolvimento, o corpo deve ser explorado primeiro. Conforme sustentam Oudeyer, Kachergis e Schueller (2019), modelos robóticos que focam nas propriedades físicas e subjetivas de um agente não são necessariamente implementados em robôs físicos. Diversos modelos são inteiramente implementados em um mundo virtual simulado com o corpo do robô, o qual é adaptado para permitir sua interação física com o meio (PFEIFER, 1997; BROOKS; STEIN, 1994; LUNGARELLA; SANDINI; PFEIFER, 2003; STOYTCHEV, 2009);

Situatividade

- **Situatividade:** ou situação de um agente no ambiente, implica que se um robô pode aprender e armazenar conhecimento por si próprio. O que este robô aprende são funções baseadas nas experiências e interações de seus sensores e atuadores (STOYTCHEV, 2009). Este princípio limita o aprendizado do agente às capacidades sensorio-motoras de seu corpo e às experiências de interações entre este agente e o ambiente. Restrições iniciais nos limites de movimento podem aumentar a adaptabilidade de agentes em desenvolvimento ao ambiente e simplificam sua interação (LUNGARELLA; BERTHOUSE, 2002; LUNGARELLA; SANDINI; PFEIFER, 2003);

Processo Incremental

- **Processo incremental:** a mente não é capaz de aprender tudo ao mesmo tempo, ela se adapta ao ambiente através de mudanças ou estágios (STOYTCHEV, 2009; LUNGARELLA; SANDINI; PFEIFER, 2003). Neste processo, novas estruturas são armazenadas com estruturas anteriores, menos completas e eficientes, e com suas expressões comportamentais. Ou seja, estruturas anteriores fornecem um histórico de habilidades e conhecimentos que podem ser reutilizados por estruturas mais novas (LUNGARELLA; SANDINI; PFEIFER, 2003);

Auto-organização de Atividades Espontâneas

- **Auto-organização de Atividades Espontâneas:** padrões estruturados ou globais podem emergir das ações espontâneas de um sistema, sem a necessidade de instruções explícitas (LUNGARELLA; SANDINI; PFEIFER, 2003). A atividade cognitiva e motora gera padrões auto-organizados de movimento, coordenados em ações produzidas, conforme propriedades físicas, corporais e do meio são exploradas (LUNGARELLA; SANDINI; PFEIFER, 2003; THELEN; SMITH, 1994);

Auto-Exploração

- **Auto-exploração:** o desenvolvimento dos sentidos decorre da exploração das consequências perceptivas de ações produzidas (THELEN; SMITH, 1994; LUNGARELLA; SANDINI; PFEIFER, 2003). Através da auto-exploração, os agentes exploram, descobrem e selecionam, dentre as soluções possíveis, aquelas mais adaptáveis e eficientes (LUNGARELLA; SANDINI; PFEIFER, 2003). A auto-exploração pode ser conceituada como um processo de auto-organização no qual novos movimentos são gerados e processos mais eficazes de interagir com as forças ambientais são explorados, descobertos e selecionados (THELEN; SMITH, 1994). A auto-exploração resulta no desenvolvimento da capacidade de prever consequências sensoriais de ações e do comportamento de objetos perante a eventos externos (MELTZOFF; MOORE, 1977; LUNGARELLA; SANDINI; PFEIFER, 2003);

- **Categorização:** capacidade de fazer distinções, de discriminar e identificar estímulos sensoriais, eventos, ações e emoções (LUNGARELLA; SANDINI; PFEIFER, 2003). Embora a percepção possa prosseguir sem a contribuição do aparato motor, a categorização perceptiva depende da interação entre os sistemas sensoriais e motores (LUNGARELLA; SANDINI; PFEIFER, 2003). Através dessa interação, dados sensoriais brutos podem ser estruturados em categorias dinâmicas e o processamento neural pode ser simplificado com a correlação de movimentos.

3.4 Projetos relacionados

Uma coletânea dos projetos da área da robótica do desenvolvimento foi apresentada por Lungarella, Sandini e Pfeifer (2003). Os autores investigaram implementações inspiradas pela psicologia do desenvolvimento que objetivaram modelar o desenvolvimento cognitivo para verificar a influência da interação experienciada pelo agente e que forneceram evidências para experimentos em robôs. Os estudos são classificados por Lungarella, Sandini e Pfeifer (2003) de acordo com seu tipo de desenvolvimento e aprendizado, através de **interação social** ou de **interação não-social**.

3.4.1 Interação social

Nestes sistemas, o desenvolvimento cognitivo está diretamente relacionado com a interação do agente com outros sujeitos e com o ambiente (MELTZOFF; MOORE, 1977; BARON-COHEN, 2001). Os robôs aprendem habilidades por meio de experiências interativas com humanos ou com outros robôs, incluindo aquisição de linguagem, atenção compartilhada e reprodução de movimentos básicos (FONG; NOURBAKHS; DAUTENHAHN, 2003). De acordo com Scassellati (2001), sistemas desenvolvidos para atuar com tarefas sociais devem responder a estados de meta, desejo e intenção.

1. A interação humano-robô foi foco de Dautenhahn e Numaoka (1999), que descreveram o surgimento de padrões de interação global através de explorações dinâmicas de movimento. Os recursos de aprendizado e comportamento dos agentes são fornecidos pela arquitetura DRAMA (*Dynamic Recurrent Associative Memory Architecture*). A estrutura permite o aprendizado de regularidades espaço-temporais e séries temporais e possui memórias de curto e longo prazo. Os experimentos realizados foram baseados nas seguintes correspondências da teoria do desenvolvimento cognitivo defendida por Vygotsky (1986. Original: 1934):
 - a) **Comunicação:** linguagem interpretada artificialmente;
 - b) **Aprendizagem em contexto social:** o robô utilizou uma estratégia de imitação, sendo guiado pelo ambiente, relacionando a proposta de Vygotsky (1986. Original: 1934) de organizar experiências. Os experimentos não especificavam o que deveria ser aprendido e os símbolos comunicados eram tratados pelo agente como entradas perceptivas;
 - c) **Internalização:** o robô aprendeu com base em suas próprias experiências sensório-motoras, com a própria atividade no centro do processo de aprendizado. Conforme o agente aprendeu os símbolos que descreviam certas situações, foi formada uma base para ser utilizada em sequências simbólicas para, por exemplo, produzir uma frase;
 - d) **Entendimento compartilhado:** a aquisição da linguagem resultou de interações sociais repetidas até que o agente tenha aprendido cada palavra com sucesso, sendo capaz de usar sua memória associativa para reproduzir as palavras em um contexto semelhante. A combinação de sinais resultou de várias associações espaço-temporais no espaço de estados dos sensores e atuadores do robô.

2. Um modelo híbrido de atenção compartilhada com mecanismos de aprendizado social que responde interações sociais básicas, em tempo real, foi desenvolvido por [Scassellati \(2001\)](#). Neste sistema, entradas visuais foram combinadas em áreas de interesse e possibilitaram ao agente selecionar os objetos mais salientes. Para cada objeto, foram produzidos indicadores como direção e orientação, utilizados pelo robô para acompanhar os objetos mais salientes com seu rosto e olhos. Objetos em movimento foram categorizados como "animados" e processados por um sistema de detecção de faces. Qualquer face detectada tornava-se foco da atenção do agente como um instrutor. Através de estratégias de imitação, o robô desenvolveu movimentos dos braços através de correlações espaciais de seu corpo com o corpo do instrutor;
3. A arquitetura SOAR é utilizada por [Laird \(2012\)](#), [Laird et al. \(2012\)](#) e [Mininger e Laird \(2016\)](#) com robôs físicos acoplados com câmera e laser, utilizados para a classificação simbólica de objetos. Os agentes foram induzidos a aprender linguagem natural através do aprendizado de tarefas como o deslocamento de objetos ou o monitoramento do ambiente. A distância percorrida foi calculada utilizando um odômetro e as informações sensoriais foram captadas por um sistema de visão espacial (SVE) e convertidas em uma cena gráfica composta de objetos discretos. Esta cena gráfica apresentava as propriedades contínuas do ambiente, que possibilitou aos agentes representarem internamente a navegação espacial. A memória de trabalho representava simbolicamente as características extraídas do ambiente através do SVE e os objetivos dos agentes. Objetos discretos integravam o ambiente e os agentes armazenavam contínuas representações internas e simbólicas. Os agentes construíram dinamicamente mapas do ambiente, com suas propriedades espaciais, assim como a localização de objetos. Através de interfaces de conversação, os objetivos foram atribuídos aos agentes, em tarefas como encontrar e entregar objetos. Foram criadas representações destas tarefas na memória semântica de cada agente e novas informações foram armazenadas nestas estruturas, como seus objetivos, objetos e agentes relacionados. Os agentes possuíam operadores modelados como regras na memória procedimental que permitiam navegar pelo ambiente, interagir com objetos e se comunicar com instrutores. Os agentes examinaram possíveis operadores com base nos objetos visíveis, em todos os objetos na memória de trabalho e na memória de longo prazo. Ao receber uma tarefa, os agentes pesquisavam em seus operadores modelados para encontrar soluções, operadores que pudessem ser aplicados de forma a concluir os objetivos. Se nenhuma solução fosse encontrada, os agentes criavam novas representações que incluíssem as restrições destas situações como referências, tornando-se as soluções para estes impasses;
4. A arquitetura SOAR foi implementada na robô Rosie ([MOHAN; LAIRD, 2014](#); [KIRK; LAIRD, 2019](#)) para o aprendizado de linguagem simbólica através de jogos de tabuleiro. A robô foi conectada a um *Kinnect* e seu sistema perceptivo possuía representações simbólicas para a identificação de objetos, suas características visuais e espaciais. Como o SOAR não apresenta módulos perceptivos, a agente foi limitada computacionalmente a aprender jogos com poucos elementos e estados. Com o estabelecimento de objetivos em linguagem natural, a agente criava representações simbólicas com base em seu conhecimento para o significado de cada sentença. As representações eram decompostas em elementos e ações que pudessem favorecer os objetivos, como a criação de suboperadores no Soar. Ao não possuir operadores aplicáveis ou encontrar ambiguidades, a robô pedia ajuda ao instrutor. Ao jogar diferentes jogos com similaridades entre as peças, a agente adquiriu diferentes significados para representações similares com contextos diferentes;

3.4.2 Interação não-social

Os estudos são caracterizados pela dependência dos processos sensoriais e motores com o ambiente e não envolvem interação com outros agentes, robôs ou humanos ([LUNGARELLA; SANDINI; PFEIFER, 2003](#)). São exploradas as

capacidades corporais dos robôs, seus efeitos na aquisição de habilidades cognitivas e motoras, e a imposição de limites ou restrições (ROCHAT; STRIANO, 2000). A coordenação sensório-motora envolve sistemas sensoriais e motores e permite a um agente induzir padrões de movimento estáveis em diferentes canais, formando associações multimodais (PFEIFER; SCHEIER, 1994). Como destaca Thelen e Smith (1994), essas associações são um pré-requisito para a formação de conceitos.

Estratégias de exploração são usadas para extrair diferentes tipos de informação do ambiente (PFEIFER; SCHEIER, 1994). Padrões estruturados podem emergir das interações locais entre os componentes que constituem um sistema, sem a necessidade de instruções explícitas ou mecanismos inatos, em uma auto-organização (LUNGARELLA; SANDINI; PFEIFER, 2003; THELEN; SMITH, 1994). O sistema de sensores e a quantidade de informações sensoriais a serem processadas devem ser equilibrados de acordo com seus objetivos e tarefas impostas (DAUTENHAHN; NU-MAOKA, 1999). Os robôs aprendem habilidades como categorização, coordenação de controles sensoriais e auto-exploração (THELEN; SMITH, 1994; PFEIFER; SCHEIER, 1994; LUNGARELLA; SANDINI; PFEIFER, 2003).

1. O quebra-cabeças da Torre de Hanói foi utilizado por Benjamin, Lyons e Lonsdale (2004) para validar a arquitetura de integração ADAPT (*Adaptive Dynamics and Active Perception for Thought*), que combina a arquitetura SOAR (LAIRD, 2008) e um modelo hierárquico de módulos sensório-motores, chamado RS. O planejamento na ADAPT é realizado através de esquemas e subesquemas, mecanismos que utilizam os operadores e suboperadores no Soar em integração com as propriedades sensório-motoras definidas no RS. Uma memória declarativa foi adicionada ao SOAR para esquemas com um conjunto de operadores que instanciavam um esquema e transformavam a instância em um conjunto de produções. O agente possuía representações internas simbólicas de cada peça, de suas características individuais e de seus possíveis posicionamentos. Os operadores permitiam ao agente criar e modificar as representações e mover as peças dos jogos entre locais pré-definidos;
2. Outros experimentos que analisam a exploração do ambiente foram propostos por Hanford, Janrathitikarn e Long (2009). Os autores utilizaram dois robôs móveis com sonar e laser, um hexápode, denominado HexCrawler, e um robô móvel com quatro rodas, denominado SuperDroid. Ambos os agentes tiveram a arquitetura SOAR implementada. As informações sensoriais foram obtidas através de módulos *BrainStem*, que possibilitam leitura e escrita de sinais digitais, e de um módulo GPS para definir e armazenar a localização dos agentes. O agente desenvolvido para o hexápode foi instruído a ir para um local definido no GPS para outro e evitar obstáculos. A arquitetura do robô HexCrawler possuía dois operadores: **i)** desviar, quando um obstáculo era detectado pelos sensores do agente; e **ii)** ir em direção ao objetivo, quando não eram detectados obstáculos. Os autores denotam que a gravação de operadores ajudou o hexápode a se afastar de um obstáculo que já havia sido evitado anteriormente. Já para o SuperDroid, Hanford, Janrathitikarn e Long (2009) descrevem que, além dos operadores aplicados no HexCrawler, foi criado um novo operador que instruía o agente a seguir paredes, o que colaborou para a navegação do robô em ambientes sem saída;
3. A arquitetura CLARION foi utilizada por Licato et al. (2015) em um agente virtual para a execução da tarefa de flutuação de Piaget (PIAGET, 1952). Conforme destaca Licato et al. (2015), experimentos piagetianos são difíceis de modelar computacionalmente por dependerem de objetos que precisam se mover de forma fisicamente realista, além da necessidade de diálogo explicativo. Na implementação de Licato et al. (2015), o agente foi posicionado abaixo de um tanque de água e tinha como objetivo identificar se um objeto iria flutuar ou afundar;
 - a) As informações sensoriais de baixo nível (massa, volume e cor) eram capturadas do ambiente simulado e alimentavam nós de entrada no nível inferior do Subsistema não-Centrado em Ações. Três definições estavam presentes como saídas no nível superior: Flutuar, Afundar e Estacionário;

- b) Um objeto com volume e peso gerados aleatoriamente era criado e aparecia no meio do tanque, onde flutuava até o topo, afundava até o fundo, ou permanecia relativamente imóvel. Após alguns segundos, este objeto desaparecia e o processo se repetia. Isso permitia que o agente coletasse dados sobre o que era observado;
 - c) Cada instância de um objeto que aparecia no mundo virtual foi registrada pelo agente e denominada exemplo. Cada recurso de entrada (massa, área, volume) era ativado por vez e esta ativação era propagada até as saídas. A saída de maior ativação era tomada como a previsão deste exemplo. Os pesos de cada item que integravam a rede eram atualizados em retropropagação a cada exemplo. O erro médio em n exemplos foi definido como a taxa de erro para o recurso isolado. O recurso que obtivesse a menor taxa de erro era levado a ser o vencedor da iteração. As iterações foram repetidas 1000 vezes por experimento;
 - d) De acordo com [Licato et al. \(2015\)](#), a seleção de recursos foi rapidamente desenvolvida nas características de massa ou volume. Novos nós foram criados para representar novos conceitos, como densidade.
4. Um modelo cognitivo para navegação, baseado na arquitetura LIDA, foi implementado em um robô humanoide, desenvolvido por [Madl et al. \(2016\)](#). O modelo possui um módulo de memória espacial, que representa o ambiente em mapas espaciais, nomeando locais e as distâncias entre eles. Este módulo é integrado com memória de trabalho, *codelets* atencionais e seleção de ações. Os endereços eram armazenados em uma estrutura de árvore ordenada de acordo com as recordações nos mapas, construindo uma hierarquia com base nos relacionamentos de endereços que são recordados em sequência ([MADL et al., 2016](#)).

3.5 *Cognits*

cognit O conceito de *cognit* é proposto por [Fuster \(2006\)](#) para representar redes de neurônios no córtex associativo que constituem as unidades básicas de memória ou conhecimento. Estas redes distribuídas são interativas, associadas pela experiência e informações adquiridas através dos sentidos. O conteúdo de um *cognit* deriva de seus componentes e das relações entre eles. Assim, o código de memória é essencialmente um código relacional. Dado que qualquer neurônio no córtex pode ser parte desta rede cognitiva, a especificidade de uma memória individual é determinada pela combinação de neurônios que a constituem.

São delineadas por [Fuster \(2006\)](#) duas hierarquias de *cognits*: perceptuais e executivos, em um esquema que considera sua organização em camadas e seus modos básicos de conectividade subjacente. Além disso, de acordo com [Fuster \(2006\)](#), todas as funções cognitivas, como percepção, atenção e memória, consistem em conexões e transações neurais entre *cognits*. A organização temporal das ações nos domínios da comportamento, raciocínio e linguagem são baseadas nas interações entre as redes corticais pré-frontais e posteriores, no topo do ciclo de percepção-ação.

Uma classe de nome *Cognit* foi implementada no CST. Esta classe é utilizada na implementação do algoritmo GLAS como Central Executiva no CST ([GUDWIN et al., 2013](#)). A classe é composta por três *codelets* integrados: i) *ActionSelectionCodelet*, que seleciona a ação mais apropriada de acordo com um Objeto de Memória de Estímulo e de um MOs da Árvore de Soluções; ii) *LearnerCodelet*, que opera paralelamente à seleção de ações e atualiza o Objeto de Memória da Árvore de Soluções dado um Objeto de Memória de Sequência de Eventos; iii) *SequenceBuilderCodelet*, que atualiza o Objeto de Memória de Sequência de Eventos na memória nos Objetos de Memória de Estímulo, Ações Selecionada e Recompensas recebidas ([RAIZER, 2015](#); [GUDWIN et al., 2013](#)).

3.6 A analogia de Berto

Tendo em vista a necessidade de padronização para investigar como mecanismos de aprendizado evoluem em robôs utilizados em estudos na área da robótica do desenvolvimento, Berto (2020) estudou funções cognitivas para identificar quais componentes seriam necessárias para validar arquiteturas cognitivas neste contexto. Com o objetivo de aproximar a teoria de Piaget aos modelos e estruturas cognitivas, Berto (2020) propôs uma analogia à classe *Cognit* do CST e aos **esquemas** propostos por Piaget no desenvolvimento do estágio sensório-motor (PIAGET, 1952; PIAGET, 1971).

O conceito de *cognit* é utilizado por Berto (2020) para representar elementos de processamento de memória com um nível de avaliação. Os *cognits* foram inseridos no contexto do estágio sensório-motor de Piaget, na seguinte caracterização:

- **1º Subestágio. Uso de Reflexos:** criação dos *cognits* inatos (ações, estados e mecanismo de aprendizado);
- **2º Subestágio. Reações Circulares Primárias:** criação e adaptação de *cognits* baseados em funções, mas não intenções.
- **3º Subestágio. Reações Circulares Secundárias:** criação e adaptação de *cognits* para formar planos, objetivos e intenções;
- **4º Subestágio. Coordenação de esquemas secundários:** reutilização e generalização de planos. Aprender o conceito de permanência de objetos;
- **5º Subestágio. Reações Circulares Terciárias:** exploração de *Affordances*. *Affordances* são qualidades de objetos que permitem identificar suas funcionalidades sem a necessidade de prévias explicações, ocorrendo intuitivamente ou baseadas em experiências anteriores (COLOMBINI, 2014). Quanto maior for a *affordance* de um objeto, melhor será a identificação de seu uso;
- **6º Subestágio. Combinações mentais:** realização de imaginação e simulação mental.

3.6.1 Experimentos Cognitivos

Com base nos estudos de Piaget (PIAGET, 1952) sobre o período sensório-motor, na Escala de Desenvolvimento Infantil de Bayley (MICHALEC, 2011), nos diferentes cenários descritos por Boyd e Bee (2011), Papalia e Feldman (2013) e Guerin, Kruger e Kraft (2013) para a avaliação do desenvolvimento do aprendizado em infantes, e nos parâmetros e níveis do ConsScale (ARRABALES, 2020), Berto (2020) propôs cenários computacionais para a incorporação de comportamentos humanos em robôs e para análise do desenvolvimento cognitivo artificial. Os experimentos são classificados de acordo com o tipo de habilidade a ser aprendida pelo agente.

Neste trabalho, são propostos os parâmetros para implementação dos experimentos de Berto (2020) relacionados com o **rastreamento de objetos**. Os robôs aprendem as habilidades de rastrear objetos através de sensores de visão. Estes experimentos, propostos por Berto (2020), são apresentados no Anexo A em uma sequência de subetapas, com cada novo subestágio se relacionando de forma incremental com os anteriores.

4 METODOLOGIA

"O homem cria a ferramenta. A ferramenta recria o homem."

(Marshall McLuhan)

Understanding Media: The Extensions of Man

Após a visita das contribuições mútuas entre os estudos da mente e das áreas de robótica e computação cognitiva, as características e implementações de agentes cognitivos artificiais e os experimentos sensório-motores para a avaliação de arquiteturas cognitivas propostos por Berto (2020), descreve-se neste capítulo: i) a abordagem proposta para a implementação deste trabalho; ii) a descrição das ferramentas utilizadas na modelagem e aplicação da arquitetura CONAIM com elementos atencionais *bottom-up* e *top-down* implementados com as ferramentas cognitivas do CST; e iii) a metodologia utilizada na definição dos experimentos.

4.1 Descrição geral

O presente trabalho propõe uma implementação da arquitetura cognitivo-atencional CONAIM com as ferramentas cognitivas do *framework* CST e um único mecanismo de aprendizado procedimental que, sem modificações externas, demonstra o processo incremental de aprendizado nos três primeiros subestágios sensório-motores na Teoria de Piaget. Este mecanismo de aprendizado permite ao modelo criar elementos em sua memória procedimental para reutilizar funções ou procedimentos que já foram aprendidos. Além disso, para cenários complexos, o mecanismo permite ao modelo combinar estes elementos em elementos novos, de forma a se adaptar a novas situações.

Para implementar a arquitetura CONAIM+CST com vistas à execução e à avaliação do conjunto de experimentos incrementais de rastreamento de objetos, proposto por (BERTO, 2020), empregaremos neste trabalho robôs móveis em ambientes simulados. Para tal, o sistema atencional da agente simulado recebe estímulos relacionados aos objetivos de cada experimento. Estes estímulos são submetidos aos sistema atencional do CONAIM composto por mecanismos *bottom-up* e *top-down*. As implementações e as formas de análise dos resultados têm como base os experimentos propostos por Berto (2020) para a aquisição de comportamentos inteligentes em agentes artificiais, a literatura relacionada (PIAGET, 1952; PIAGET, 1999; TULVING, 1985; LUNGARELLA; SANDINI; PFEIFER, 2003; PAPALIA; FELDMAN, 2013) e adaptações derivadas de Araújo e Gudwin (2018), Regattieri e Colombini (2018), Berto et al. (2020b) e Berto (2020). Um ambiente de simulação é utilizado para realizar os experimentos cognitivos e validar a arquitetura.

4.2 Materiais

4.2.1 Ambiente computacional

O ambiente para simulação de agentes robóticos escolhido para realização dos experimentos foi o CoppeliaSim, uma atualização do simulador V-REP (ROHMER; SINGH; FREESE, 2013). O CoppeliaSim é utilizado para o desenvolvimento de algoritmos, simulações de automação, prototipagem, educação relacionada à robótica e monitoramento remoto (ROHMER; FREESE, 2019). O simulador CoppeliaSim: i) É gratuito; ii) Permite a aceleração da simulação; iii) É compatível com diferentes linguagens computacionais, como C / C ++, Python, Java, Lua, Matlab e Octave; iv) É cross-plataforma; v) Possui um ambiente de desenvolvimento integrado; vi) É baseado em uma arquitetura de controle distribuído.

No simulador, cada objeto ou modelo pode ser controlado individualmente por meio de um *script* incorporado, como um cliente remoto via RemoteAPI. Os agentes foram modelados no ambiente de simulações CoppeliaSim. De

forma a promover uma simulação mais próxima da realidade, serão empregados modelos de erro aos sensores destes robôs. Os experimentos foram conduzidos em um notebook com processador Intel Core i5 2.7GHz, sistema operacional Windows 10, com 8GB de memória RAM e placa de vídeo Nvidia GeForce 940MX.

4.2.2 Agentes Robóticos

A **humanoide Marta**, apresentada na Figura 11(a), foi originalmente modelada utilizando o software CAD SolidWorks. Seu modelo foi simplificado e importado para o simulador *CoppeliaSim*, conforme apresentado na figura 11(b). Foram estabelecidas as propriedades dinâmicas para o robô real, derivando um modelo para sua simulação dinâmica, com as medidas das massas das peças impressas e a geometria das malhas sendo utilizados para calcular os centros de inércia e massa do robô. A humanoide possui 110 cm de altura e 25 graus de liberdade. Os graus de liberdade da humanoide são ilustrados na Figura 11(c). O limite de torque foi configurado para simulação com base na especificação dos atuadores reais da robô (25 N.m). Também foi inserido um sensor de visão, posicionado na altura de sua cabeça, deslocado 1.99m no eixo Y, acima do chão, e inclinado 20° para baixo no eixo X, conforme demonstra a Figura 12.

humanoide
Marta

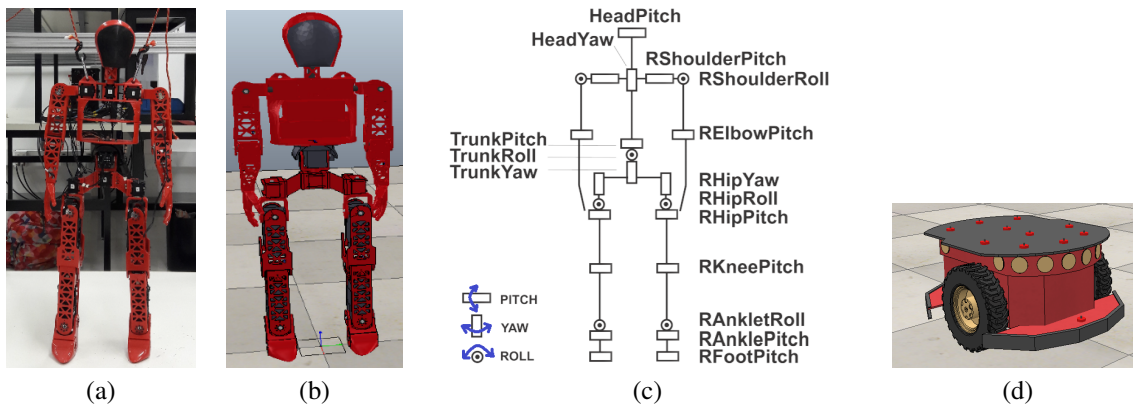


Figura 11 – **a)** Humanoide Marta no Laboratório de Automação e Sistemas Integráveis (LASI) - UNESP Sorocaba/SP. **b)** Humanoide Marta no Simulador CoppeliaSim. **c)** 25 Graus de Liberdade da humanoide. Fonte: (TOMAZELA; COLOMBINI, 2019). **d)** Robô Pioneer P3DX no Simulador CoppeliaSim.

O robô **Pioneer P3DX**, já disponível no simulador CoppeliaSim, possui 2 motores independentes, um para cada roda, e 16 sensores de proximidade, 8 em sua frente dispostos em 180° e os outros 8 na traseira também dispostos em 180°. Esses sensores dependem da propagação ultrassônica para produzir dados. Este sensor é descrito em Rohmer e Freese (2019) como um sensor de alta resolução sem colisão para a coleta de informações de distância de obstáculos.

Pioneer P3DX

O modelo Pioneer P3DX no simulador CoppeliaSim possui um Algoritmo de Braitenberg (BRAITENBERG, 1984) disponível. O algoritmo é baseado no mundo animal de maneira minimalista e construtiva, com comportamentos relativos simples e espaciais. O movimento é controlado diretamente pelos sensores de proximidade, afastando-o de obstáculos ou outros agentes (ROHMER; FREESE, 2019). Durante a execução dos experimentos cognitivos propostos para este projeto, este robô não terá o algoritmo cognitivo-atencional implementado, somente o algoritmo de Braitenberg, comportando-se como um corpo móvel no ambiente. A movimentação do robô Pioneer foi utilizada neste trabalho para a investigação do comportamento do algoritmo cognitivo atencional implementado na humanoide Marta com objetos móveis, para aprendizado das tarefas relacionadas ao rastreamento de objetos.

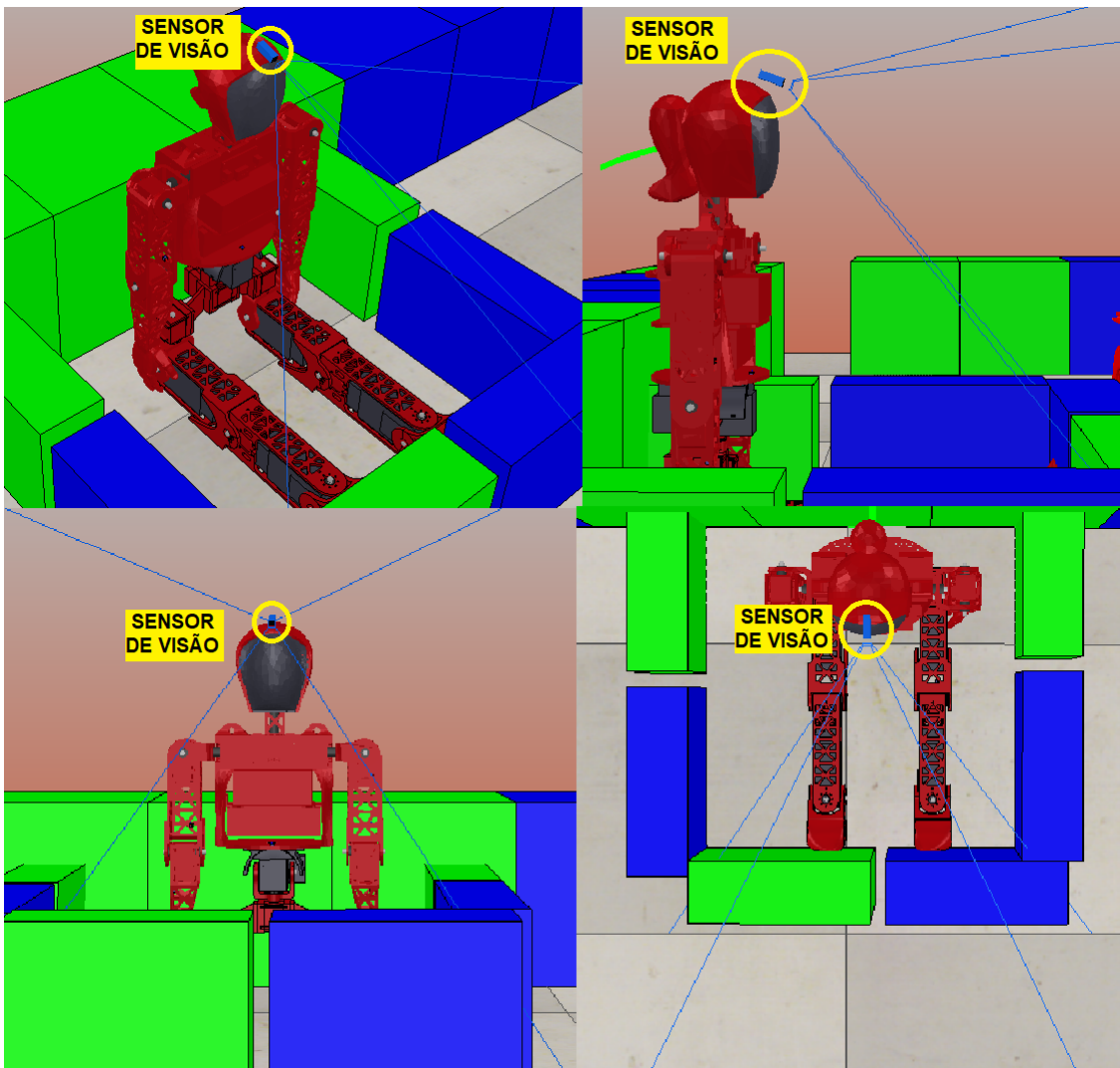


Figura 12 – Posição do Sensor de Visão da Humanoide Marta no Simulador CoppeliaSim. Y: 1.99m e rotação em X: -20°.

4.3 Modelagem da agente cognitivo-atencional segundo o CONAIM

4.3.1 Espaço Sensorial

O simulador CoppeliaSim (ROHMER; FREESE, 2019; ROHMER; SINGH; FREESE, 2013) possibilita a utilização de sensores de visão. Os sensores de visão, objetos visíveis, operam de maneira semelhante aos objetos de câmera: os objetos que estão em seu campo de visão são renderizados e dispararão a detecção se os limites especificados forem disparados acima ou abaixo. Sensores de visão, que podem detectar entidades renderizáveis, devem ser usados sobre sensores de proximidade principalmente quando a cor, luz ou estrutura desempenham um papel no processo de detecção (por exemplo, sensores infravermelhos ou, mais geralmente, sensores sensíveis à luz (câmeras, etc.)) . No entanto, dependendo da placa gráfica em que o aplicativo está sendo executado ou da complexidade dos objetos da cena, os sensores de visão podem ser um pouco mais lentos do que os sensores de proximidade.

Neste projeto, a percepção será realizada sobre o sensor de visão da humanoide Marta. O sensor possui uma câmera perspectiva. Conforme ilustra a Figura 13, cada coordenada 3D (x, y, z) do CoppeliaSim é projetada em coordenadas

2D (u, v) em um plano de imagem (Π) com raios em perspectiva, originados no centro de projeção (O). A origem do sistema de coordenadas é considerada como sendo a origem (O) e a distância focal (f) é a distância da origem ao plano, ao longo do eixo principal (ou eixo óptico). Utilizou-se uma distância focal de 3cm. O eixo óptico é alinhado com o eixo $\hat{z}$. O sensor disponibiliza os valores dos canais RGB e a medida de distância relativa a partir do centro de projeção (O) do sensor. Este sensor de visão do CoppeliaSim é análogo a um sensor RGB-D.

As informações obtidas pelo sensor são enviadas através da RemoteAPI, que permite a comunicação entre um cliente (agente cognitivo) e o servidor (CoppeliaSim). As informações do sensor de visão são recebidas pelo algoritmo implementado em um vetor de bytes. A imagem resultante deste vetor é uma cópia da imagem original do sensor invertida verticalmente, ilustrada na Figura 13 (a). A resolução deste sensor é configurável. Propomos a utilização de uma resolução de 65.536 pixels (256 pixels x 256 pixels), em que o número de pixels p por eixo é 256 ($p=256$ pixels). Assim, o *array* recebido é constituído de 196.608 pixels (65.536 pixels em 3 canais: R, G e B) na forma $Value_{color}(x, y)$, com x e y representando a posição horizontal e vertical do pixel na imagem, sendo:

$$Array_{visao} = [Value_{Red}(0,0), Value_{Green}(0,0), Value_{Blue}(0,0), \dots, Value_{Red}(255,255), Value_{Green}(255,255), Value_{Blue}(255,255)], \text{ e } x \in [0, p-1] \text{ e } y \in [0, p-1].$$

São utilizadas técnicas de redução de resolução para simular os diferentes níveis de acuidade visual ao longo do desenvolvimento humano. Esse ganho incremental de qualidade sensorial permite ao modelo adaptar o aprendizado motor de forma gradual. Assim, mecanismos mais simples são aprendidos e reutilizados para lidar com ambientes mais complexos.

A partir do *array visão* recebido, para cada canal RGB, são divididos e armazenados temporariamente dados sensoriais como observações: 65.536 leituras do sensor de visão em cada um dos canais RGB (*Red-Green-Blue*). Os estados de observação para a humanoide Marta são definidos como:

- O_R : Observação das leituras do canal da cor vermelha (R - *Red*), definida por $o_{R_{n_t}} = vision_{red_{n_t}}$ com $n \in [1, 65.536]$;
- O_G : Observação das leituras do canal da cor verde (G - *Verde*), definida por $o_{G_{n_t}} = vision_{green_{n_t}}$ com $n \in [1, 65.536]$;
- O_B : Observação das leituras do canal da cor azul (B - *Blue*), definida por $o_{B_{n_t}} = vision_{blue_{n_t}}$ com $n \in [1, 65.536]$.

Através do sensor de visão, também é possível obter um array de 65.536 pixels (256 pixels x 256 pixels) com medidas relativas à distância do plano de imagem (Π) do sensor de visão aos obstáculos. Esses dados são armazenados temporariamente como observações: 65.536 leituras de distâncias relativas ao plano de imagem (Π) do sensor de visão, com valores em uma faixa de 0 (mais próximo) a 10 (mais distante). Os estados de observação para a humanoide Marta são definidos como:

- O_D : Observação das leituras de distância relativa ao sensor de visão, definida por $o_{D_{n_t}} = vision_{depth_{n_t}}$ com $n \in [1, 65.536]$;

4.3.2 Extração de Características *bottom-up*

O modelo atencional proposto pelo CONAIM é capaz de combinar diferentes dimensões de características fornecidas por múltiplos sensores. Neste trabalho, os experimentos cognitivos serão realizados com a **humanoide Marta** e os sensores embarcados na mesma, descritos na seção anterior. A partir dos dados coletados dos canais de cores (RGB) nos espaços de observação do sensor de visão são construídos três Mapas de Características *bottom-up*, um para cada canal RGB. Os mapas *bottom-up* relacionados aos canais de cores são gerados com as médias dos valores obtidos pelo

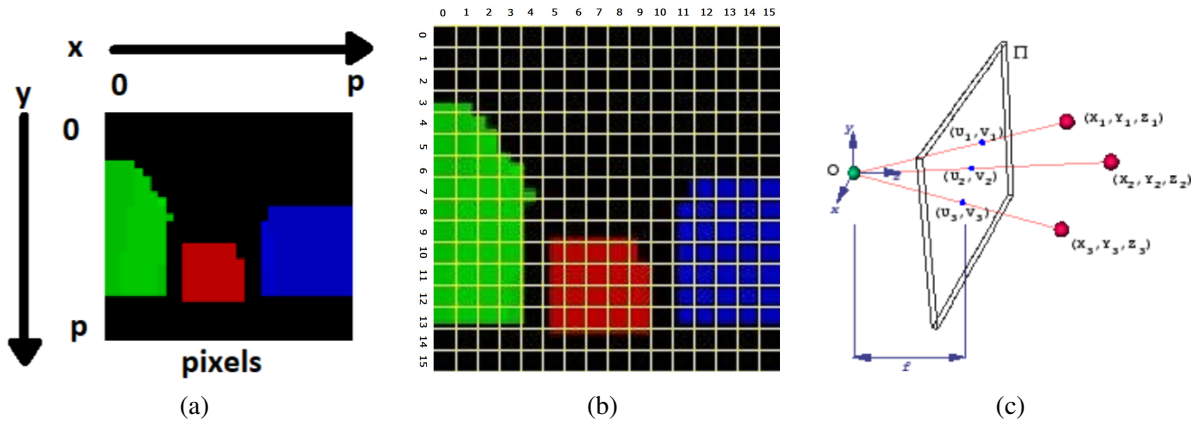


Figura 13 – **a)** Imagem recebida através da RemoteAPI. Os eixos representam as coordenadas x e y de cada pixel. **b)** 256 regiões utilizadas no cálculo dos mapas de características para este trabalho. **(c)** Ilustração da conversão 3D para 2D.

sensor de visão em r regiões pré-definidas, como ilustrado na Figura 13 (b). Neste trabalho foram utilizadas $r=256$ regiões. O número de regiões pode ser alterado de acordo com a precisão desejada. São obtidos 256 valores para representar cada um dos mapas de características *bottom-up*, em correspondência a cada uma das 256 regiões. As *features bottom-up* extraídas a partir dessas observações são:

- $\mathcal{F}_1$. **Canal Red.** Representa a média dos valores das medidas do sensor de visão para o canal *Red* (Vermelho) para cada região da Figura 13 (b). A função de mapeamentos de características *bottom-up* ϕ_{BU_1} é responsável por extrair informações sobre os níveis da cor vermelha nos elementos dentro do campo de visão da agente atencional. O processo para cálculo de cada elemento f_{1_i} do mapa de características $\mathcal{F}_1$, delimitado a uma região r , é demonstrado através da aquisição do array $Mean[]$ no Algoritmo 1 com os níveis de cor para o canal Vermelho. O array $Mean[]$ representa o valor calculado para determinado elemento f_{1_i} do mapa de características $\mathcal{F}_1$. O processo também é ilustrado de uma forma simplificada na Figura 14 (a);
- $\mathcal{F}_2$. **Canal Green.** Representa a média dos valores das medidas do sensor de visão para o canal *Green* (Verde) para cada região da Figura 13 (b). A função de mapeamentos de características *bottom-up* ϕ_{BU_2} é responsável por extrair informações sobre os níveis da cor verde nos elementos dentro do campo de visão da agente atencional. O processo para cálculo de cada elemento f_{2_i} do mapa de características $\mathcal{F}_2$, delimitado a uma região r , demonstrado através da aquisição do array $Mean[]$ no Algoritmo 1 com os níveis de cor para o canal Verde. O array $Mean[]$ representa o valor calculado para determinado elemento f_{2_i} do mapa de características $\mathcal{F}_2$. O processo também é ilustrado de uma forma simplificada na Figura 14 (a);
- $\mathcal{F}_3$. **Canal Blue.** Representa a média dos valores das medidas do sensor de visão para o canal *Blue* (Azul) para cada região da Figura 13 (b). A função de mapeamentos de características *bottom-up* ϕ_{BU_3} é responsável por extrair informações sobre os níveis da cor azul nos elementos dentro do campo de visão da agente atencional. O processo para cálculo de cada elemento f_{3_i} do mapa de características $\mathcal{F}_3$, delimitado a uma região r , é obtido através da aquisição do array $Mean[]$ no Algoritmo 1 com os níveis de cor para o canal Azul. O array $Mean[]$ representa o valor calculado para determinado elemento f_{3_i} do mapa de características $\mathcal{F}_3$. O processo também é ilustrado de uma forma simplificada na Figura 14 (a);

A partir dos dados coletados da distância relativa ao plano de imagem (Π) do sensor de visão, é construído um Mapa de Características *bottom-up* de Distância. O mapa *bottom-up* relacionado à distância é gerado a partir da proximidade

detectada nas médias dos valores obtidos, de acordo com as r regiões pré-definidas. Cada uma destas regiões representa uma característica (*feature*) *bottom-up* no mapa de características (*Feature Map*) de distância. O processo também é ilustrado de uma forma simplificada na Figura 14

- $\mathcal{F}_4$. **Distância.** Representa a proximidade média a partir dos valores das medidas de distância relativa entre obstáculos e corpos móveis e a agente atencional, com valores em uma faixa de 0 (mais distante) a 1 (mais próximo). A função de mapeamentos de características *bottom-up* ϕ_{BU_4} é responsável por extrair informações sobre a disposição de elementos ao redor da agente atencional. O processo para cálculo de cada elemento f_{4_t} do mapa de características $\mathcal{F}_4$, delimitado a uma região r , é demonstrado através da aquisição do array $Mean[]$ no Algoritmo 1. O array $Mean[]$ representa o valor calculado para determinado elemento f_{4_t} do mapa de características $\mathcal{F}_4$. O processo também é ilustrado de uma forma simplificada na Figura 14 (a);

Algorithm 1 Cálculo do *bottom-up Feature Map* i em cada região da Figura 13 (b).

```

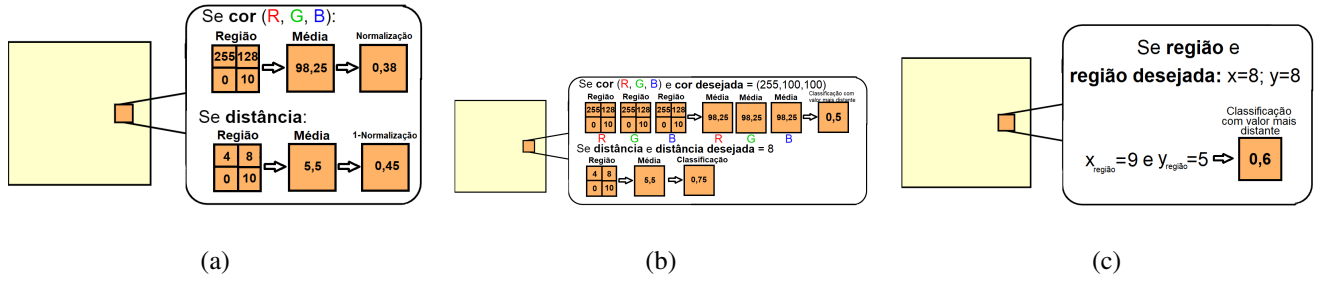
1: procedure BOTTOM-UP FEATURE MAP(Array[], type)
2:    $r \leftarrow 256$  ▷ número de regiões
3:    $div \leftarrow 16$  ▷ número de divisões em cada eixo
4:    $p \leftarrow 256$  ▷ número de pixels em cada eixo
5:    $SumRegion \leftarrow 0$ 
6:   for  $r_y \leftarrow 0$  to  $div$  do
7:      $y_i \leftarrow r_y * p / div$ 
8:      $y_o \leftarrow (1 + r_y) * p / div$ 
9:     for  $r_x \leftarrow 0$  to  $div$  do
10:       $x_i \leftarrow r_x * p / div$ 
11:       $x_o \leftarrow (1 + r_x) * p / div$ 
12:      for  $y \leftarrow y_i$  to  $y_o$  do
13:        for  $x \leftarrow x_i$  to  $x_o$  do
14:           $value \leftarrow Array[x + y * p]$ 
15:           $SumRegion \leftarrow SumRegion + value$  ▷  $f_{i_{r_x+r_y*div_t}} = Mean[r_x + r_y * div]$ 
16:        if (type == 'color'):  $Mean[r_x + r_y * div] \leftarrow SumRegion * r / (p * p)$ 
17:        if (type == 'distance'):  $Mean[r_x + r_y * div] \leftarrow 1 - SumRegion * r / (p * p)$ 
18:       $SumRegion \leftarrow 0$ 
19:   return (Mean[])

```

4.3.3 Extração de Características *top-down*

O CONAIM também é capaz de combinar características fornecidas por influências endógenas (*top-down*). Nos experimentos a serem realizados com a **humanoide Marta**, serão construídos cinco mapas de características *top-down* a partir de interpretações dos estímulos gerados pelo sensor de visão, nas r regiões pré-definidas, como ilustrado na Figura 13 (b). Cada uma destas regiões representa uma característica (*feature*) no mapa de características *top-down* (*Feature Map*) correspondente. Neste trabalho serão utilizadas $r=256$ regiões. O número de regiões pode ser alterado de acordo com a precisão desejada.

- $\mathcal{F}_5$. **Cor-Objetivo.** Esta característica representa as regiões em que uma cor desejada (com níveis definidos nos três canais RGB) é destacada e para as quais a atenção da agente pode ser direcionada, com relação às medidas do sensor de visão em cada região da Figura 13 (b). A função de mapeamentos de características *top-down* ϕ_{TD_5} é responsável por extrair informações sobre a disposição de elementos ao redor da agente atencional. Cada elemento $f_{5_{nt}}$ do mapa de características $\mathcal{F}_5$, delimitado a uma determinada região r , é computado de acordo com



a aquisição do array `OutputArray[]` no Algoritmo 2. O array `OutputArray[]` representa o valor para determinado elemento f_{5_i} do mapa de características $\mathcal{F}_5$;

- **$\mathcal{F}_6$. Distancia-Objetivo.** Esta característica representa as regiões em que uma distância desejada é destacada e para as quais a atenção da agente pode ser direcionada, com relação às medidas de distância relativa ao sensor de visão em cada região da Figura 13 (b). A função de mapeamentos de características *top-down* ϕ_{TD_6} é responsável por extrair informações sobre a disposição de elementos ao redor da agente atencional. Cada elemento $f_{6_{n_i}}$ do mapa de características $\mathcal{F}_6$, delimitado a uma determinada região r , é computado de acordo com a aquisição do array `OutputArray[]` no Algoritmo 2. O array `OutputArray[]` representa o valor para determinado elemento f_{6_i} do mapa de características $\mathcal{F}_6$;
- **$\mathcal{F}_7$. Região-Objetivo.** Esta característica representa as regiões do espaço (Figura 13 (b)) para as quais a atenção da agente pode ser direcionada, independentemente do que esteja acontecendo naquela região. A função de mapeamentos de características *top-down* ϕ_{TD_7} é responsável por extrair informações sobre a disposição de elementos ao redor da agente atencional. Cada elemento $f_{7_{n_i}}$ do mapa de características $\mathcal{F}_7$, delimitado a uma determinada região r , é computado de acordo com a aquisição do array `OutputArray[]` no Algoritmo 3. O array `OutputArray[]` representa o valor para determinado elemento f_{7_i} do mapa de características $\mathcal{F}_7$.

4.3.4 Mapa de Características Combinado

O sinal composto c_{i_t} de todos os mapas de características gerados (*bottom-up* e *top-down*) é obtido de acordo com o Modelo de Colombini (2014), através da somatória dos produtos entre um elemento do vetor de pesos de características no sistema atencional ($w_{f_{j_t}}$) e determinado mapa de características ($\mathcal{F}_{i_{j_t}}$): $c_{i_t} = \sum_{j=1}^r w_{f_{j_t}} \cdot \mathcal{F}_{i_{j_t}}$

De forma a classificar o tipo da característica vencedora do ciclo (*bottom-up* ou *top-down*) antes da aplicação do algoritmo de inibição de retorno, são obtidos dois valores durante a formação do Mapa de Características Combinado:

Algorithm 2 Cálculo do *top-down Feature Map* i em cada região da Figura 13 (b).

```

1: procedure TOP-DOWN FEATURE MAP(InputArray[])
2:    $r \leftarrow 256$  ▷ número de regiões
3:    $div \leftarrow 16$  ▷ número de divisões em cada eixo
4:    $p \leftarrow 256$  ▷ número de pixels em cada eixo
5:    $vd \leftarrow goal$  ▷ valor desejado para a medida
6:    $mr \leftarrow saturation$  ▷ maior valor possível para a medida
7:    $SumRegion \leftarrow 0$ 
8:   for  $r_y \leftarrow 0$  to  $div$  do
9:      $y_i \leftarrow r_y * p / div$ 
10:     $y_o \leftarrow (1 + r_y) * p / div$ 
11:    for  $r_x \leftarrow 0$  to  $div$  do
12:       $x_i \leftarrow r_x * p / div$ 
13:       $x_o \leftarrow (1 + r_x) * p / div$ 
14:      for  $y \leftarrow y_i$  to  $y_o$  do
15:        for  $x \leftarrow x_i$  to  $x_o$  do
16:           $value \leftarrow InputArray[x + y * p]$ 
17:           $SumRegion \leftarrow SumRegion + value$ 
18:           $Mean[r_x + r_y * div] \leftarrow SumRegion * r / (p * p)$ 
19:           $if((Mean[r_x + r_y * div] - vd) / mr \leq 20\%) : td_{value} \leftarrow 1$ 
20:           $elif((Mean[r_x + r_y * div] - vd) / mr \leq 40\%) : td_{value} \leftarrow 0.75$ 
21:           $elif((Mean[r_x + r_y * div] - vd) / mr \leq 60\%) : td_{value} \leftarrow 0.5$ 
22:           $elif((Mean[r_x + r_y * div] - vd) / mr \leq 80\%) : td_{value} \leftarrow 0.25$ 
23:           $else : td_{value} \leftarrow 0$ 
24:           $SumRegion \leftarrow 0$ 
25:           $OutputArray.insert(td_{value})$ 
return(OutputArray)

```

Algorithm 3 Cálculo do *top-down Feature Map* de Região em cada região da Figura 13 (b).

```

1: procedure TOP-DOWN FEATURE MAP(InputArray[])
2:    $r \leftarrow 256$  ▷ número de regiões
3:    $div \leftarrow 16$  ▷ número de divisões em cada eixo
4:    $p \leftarrow 256$  ▷ número de pixels em cada eixo
5:    $xd \leftarrow goal_x$  ▷ coordenada x da região desejada
6:    $yd \leftarrow goal_y$  ▷ coordenada y da região desejada
7:   for  $r_y \leftarrow 0$  to  $div$  do
8:      $y_i \leftarrow r_y * p / div$ 
9:      $y_o \leftarrow (1 + r_y) * p / div$ 
10:    for  $r_x \leftarrow 0$  to  $div$  do
11:       $if(r_x == xd \text{ and } r_y == yd) : td_{value} \leftarrow 1$ 
12:       $elif(r_x - xd \leq 2 \text{ and } r_y - yd \leq 2) : td_{value} \leftarrow 0.8$ 
13:       $elif(r_x - xd \leq 4 \text{ and } r_y - yd \leq 4) : td_{value} \leftarrow 0.6$ 
14:       $elif(r_x - xd \leq 6 \text{ and } r_y - yd \leq 6) : td_{value} \leftarrow 0.4$ 
15:       $elif(r_x - xd \leq 8 \text{ and } r_y - yd \leq 8) : td_{value} \leftarrow 0.2$ 
16:       $else : td_{value} \leftarrow 0$ 
17:       $OutputArray.insert(td_{value})$ 
return(OutputArray)

```

(i) a soma dos valores gerados pelos mapas de características *bottom-up* e (ii) a soma dos valores gerados pelos mapas de características *top-down*. Se (i) > (ii), o vencedor do ciclo atencional é classificado como *bottom-up*. Se (i) < (ii), o vencedor do ciclo atencional é classificado como *top-down*.

4.4 Um modelo de aprendizado procedimental no sistema cognitivo-atencional CONAIM+CST

A implementação do sistema cognitivo-atencional CONAIM com as ferramentas do CST (ARAÚJO; GUDWIN, 2018; REGATTIERI; COLOMBINI, 2018; BERTO, 2020; BERTO et al., 2020b) possui elementos que podem ser correlacionados com a classe *Cognit* do CST (GUDWIN et al., 2013). A classe *cognit* é formada pelo *codelet Learner* e pelos MOs Lista de Ações (*ActionList MO*) e Lista de Estados (*StatesList MO*), conforme demonstrado no esquema correspondente na Figura 10. As ações e recompensas estão relacionadas com as características de cada subestágio de Piaget levantadas por Berto (2020) e com as tarefas que os agentes desenvolverão durante o aprendizado, podendo diferir entre os experimentos.

Os estados possíveis estão diretamente vinculados ao mapa de saliências $\mathcal{L}_t$. Neste trabalho, o mapa de saliências tem 256 dimensões, relacionadas ao sensor utilizado. Devido a limitações da máquina utilizada para executar os experimentos propostos, estas 256 dimensões foram transformadas em $t=16$ dimensões, representadas pelo conjunto de dimensões D_t . As localizações destas dimensões D_t são demonstradas na Figura 15. O valor adotado para cada uma destas regiões é definido utilizando o valor máximo de cada região correspondente do mapa de saliências. Cada elemento $d_i \in \mathcal{D}_t$ é normalizado entre 0 e 1. $i \in [1, t-1]$. Após a normalização, cada elemento é classificado em $\tau = 2$ possíveis faixas. É adotado $\lambda = 1/\tau = 0,5$, dando origem a multiplicadores correspondentes $n_i \in N_t$. Os produtos de cada multiplicador $n_i \in N_t$ com o respectivo τ^i são utilizados em uma somatória, retornando o conjunto de estados E_t , conforme apresentado no processo ilustrado na Figura 9.

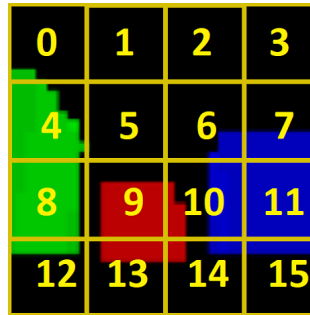


Figura 15 – 16 regiões utilizadas como estados para o Aprendizado Procedimental neste trabalho.

O conjunto de estados E_t , representado pelo *StatesList MO*, e o conjunto das possíveis ações da agente A_t , representado pelo *ActionList MO*, são utilizados no *Learner Codelet* para a criação da tabela de aprendizado QTable e, posteriormente, com as recompensas obtidas a cada ciclo, para a atualização da QTable. Após a atualização, a próxima ação é selecionada através da tabela QTable para aplicação no estado atual. Cada ação selecionada possui um *Behavioral Codelet* correspondente, que avalia a possível utilização ação com base em seu comportamento e o instante atual. As ações e suas avaliações são então enviadas para o *Memory Container*, que salva os valores na memória motora e decide qual ação deve ser executada pelo *Motor Codelet* (BERTO, 2020; BERTO et al., 2020b).

4.5 Modelo Atencional Top-Down+Bottom-up

Para validar a modelagem proposta, as classes do CONAIM *bottom-up attentional* foram adaptadas para a atuação *bottom-up e top-down attentional* com o sistema de Aprendizizado Procedimental da agente Marta, utilizando o CST. O ciclo cognitivo da arquitetura é ilustrado no esquema da Figura 16. As classes que efetuam a coleta de dados sensoriais e os mecanismos que compõem o sistema atencional do CONAIM foram implementadas para a agente com as ferramentas cognitivas do CST em linguagem *Java*. O diagrama de classes pode ser visto na Figura 17. As estruturas dessas classes herdam as características da classe *Codelet* do CST e possuem métodos abstratos. Destes métodos, o que mais se destaca é *proc()*, processo lógico e função principal da classe *Codelet*.

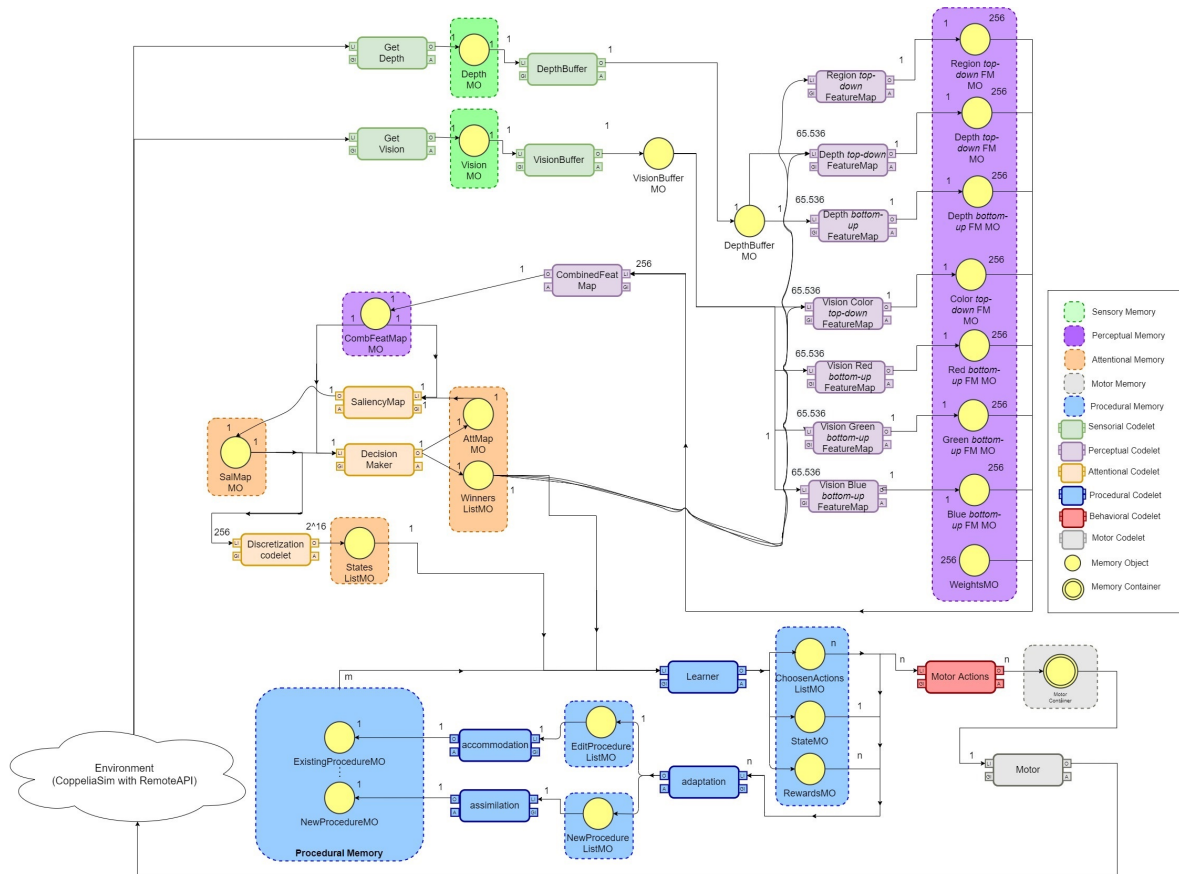


Figura 16 – Esquema de implementação da arquitetura CONAIM+CST com a humanoide Marta e que recebe estímulos atencionais *bottom-up e top-down*.

Os componentes principais são as classes *AgentMind* e *OutsideCommunication*, adaptadas do tutorial disponível em (GUDWIN et al., 2013) e das classes implementadas nos trabalhos Berto et al. (2020b), Berto (2020), Regattieri e Colombini (2018) e Araújo e Gudwin (2018). O *AgentMind* é herdado da classe *Mind*, do CST, e agrega *Raw Memory* e *Coderack* para formar o sistema cognitivo. O *AgentMind* será o criador de todos os objetos de memória (MOs) e *codelets* utilizados. Por sua vez, a classe *OutsideCommunication* efetua a comunicação externa com a robô através do RemoteAPI do simulador CoppeliaSim. Para o *SensorBufferCodelet*, foi implementado um método genérico para "cópias profundas" (*deep copies*, termo utilizado para uma cópia criada em um local diferente da memória) dos MOs relacionados aos dados sensoriais Berto (2020). Essas cópias são feitas através de objetos serializadores em linguagem Java. Como a classe *MemoryObject*, existente no CST, já implementa essa interface, as informações e

funções armazenadas em um MO instanciado também a implementam.

Os *codelets* mais primitivos da humanoide Marta são representados por *Get Vision* e *Get Depth*. Os componentes são conectados ao remoteAPI do simulador CoppeliaSim e se comunicam diretamente com o sensor de visão da agente, recebendo a imagem em um vetor de *bytes*. Estes *codelets* armazenam as informações do sensor de visão em objetos de memória *VisionMO* e *DepthMO*. O *codelet VisionBuffer* armazena os objetos de memória *VisionMO* em um objeto de memória *VisionBufferMO*. O *codelet DepthBuffer* armazena os objetos de memória *DepthMO* em um objeto de memória *DepthBufferMO*.

Por meio dos *codelets VisionRedFeatureMapCodelet*, *VisionGreenFeatureMapCodelet*, *VisionBlueFeatureMapCodelet* e *DepthBottomUpFeatureMapCodelet*, são construídos os mapas de características *bottom-up* desejados para a humanoide Marta a partir dos objetos de memória *VisionBufferMO* e *DepthBufferMO*. Estes mapas de características são armazenados em objetos de memória *RedFM*, *GreenFM*, *BlueFM* e *DepthBUFM*. Por meio dos *codelets VisionColorFeatureMapCodelet*, *RegionFeatureMapCodelet* e *DepthTopDownFeatureMapCodelet*, são construídos os mapas de características *top-down* desejados para a humanoide Marta a partir dos objetos de memória *VisionBufferMO* e *DepthBufferMO*. Estes mapas de características são armazenados em objetos de memória *ColorFM*, *RegionFM* e *DepthTDFM*. Os mapas *bottom-up* e *top-down* competem pela atenção. Seus objetos de memória são combinados em um mapa de características combinado pelo *codelet AppCombFeatMapCodelet*, conforme proposto pelo Modelo de Colombini (2014).

As funções que implementam o mecanismo de IOR, apresentadas por Colombini (2014), são baseadas nas exponenciais negativas temporais para gerar uma deterioração da atenção no tempo. Estes mecanismos são regulados por parâmetros constantes que influenciam o curso excitatório ou inibitório da atenção na operação *bottom-up* do CONAIM (COLOMBINI; SIMÕES; RIBEIRO, 2017). O mapa de saliências é obtido pelo *SaliencyMapCodelet* através do produto entre o mapa de características combinado e o mapa atencional.

O mapa atencional é inicializado com todos os seus elementos sendo iguais a 1 e é realimentado a cada iteração pelo *codelet DecisionMakingCodelet* (COLOMBINI, 2014; COLOMBINI; SIMÕES; RIBEIRO, 2017). O mapa de saliências influencia o *codelet DecisionMakingCodelet* que é implementado segundo uma abordagem *Winner-Takes-All*, que define a característica vencedora de cada ciclo atencional com base nos ciclos inibitórios e excitatórios e promove a alteração do mapa atencional. As características vencedoras *top-down* são utilizadas nos experimentos do 3º subestágio para definição das características desejadas nos mapas de características *top-down* para o próximo ciclo atencional.

O *codelet Learner* é utilizado na aplicação do algoritmo de aprendizado. A classe *QLearning*, disponível no CST, proporciona a criação e atualização de um mapa que relaciona estados, ações escolhidas para execução e as recompensas obtidas pela agente. Os estados (*StatesListMO*) são adquiridos com o mapa de saliências redimensionado e discretizado. Todas as ações possíveis para a agente integram o *memory container AllProceduresListMO* e são definidas de acordo cada subestágio, em cada lote de experimentos. As características vencedoras de cada ciclo atencional (*WinnersListMO*) são utilizadas para classificação do desempenho da agente e definição da recompensa a ser registrada no procedimento anterior.

É adotado como conceito de **esquema / procedimento (procedure)** todo objeto de memória que foi inserido na Memória Procedimental através de um *cognit* de Aprendizado Procedimental para a investigação de uma determinada ação. A cada ciclo, é verificado se um procedimento da Memória Procedimental pode ser executado para o estado atual. Caso diferentes ações estejam relacionadas a um mesmo estado, elas são executadas em sequência decrescente, de acordo com suas recompensas. Os procedimentos existentes que forem novamente executados são atualizados com suas recompensas adquiridas.

Novos procedimentos podem ser aprendidos e procedimentos já conhecidos podem ser refinados. O estado, as ações escolhidas para execução e as recompensas obtidas pela agente na avaliação do algoritmo de aprendizado são

4.6 Experimentos propostos

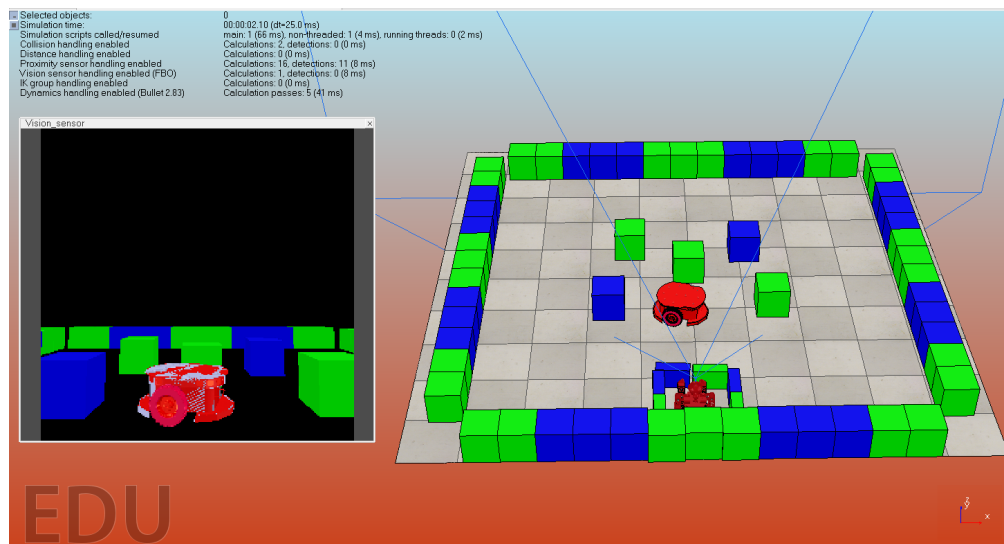
Baseando-se na analogia de Berto (2020) com a classe *cognit* do CST e em seus experimentos propostos para a análise do desenvolvimento cognitivo artificial em cenários computacionais, nas descrições das características do subestágio sensorio-motor de Piaget (1952), Chec e Martin (2002), Wood, Smith e Grossniklaus (2011), Cole e Cole (2003), Wadsworth (1997), Piaget (1999), Piaget (1971) e Papalia e Feldman (2013), nas descrições do desenvolvimento de memórias de Tulving (1985) e Armstrong et al. (2014) e nas principais componentes presentes nas arquiteturas cognitivas, são propostos experimentos para a investigação do comportamento inteligente da arquitetura cognitiva CONAIM+CST implementada com a agente Marta.

São propostos três experimentos, correspondentes aos primeiros três subestágios do período sensorio-motor de Piaget. Cada experimento possui uma estrutura baseada no modelo CONAIM+CST, conforme ilustrado na Figura 19. A estrutura do modelo é incrementada com a inserção de novos elementos a cada experimento e subestágio correspondente. O mecanismo de aprendizado do modelo é inalterado. Os procedimentos $m_p \in M_p$, são criados como objetos de memória no CST, e são análogos aos esquemas propostos de Piaget. Cada procedimento/esquema representa a atuação de uma ação em um estado e sua recompensa. Um *cognit* de L_p atua sobre M_W , atualizando o conteúdo do $m_p \in M_p$, através de *codelets* de assimilação e acomodação. Portanto, a implementação proposta atua como uma máquina de processamento de esquemas.

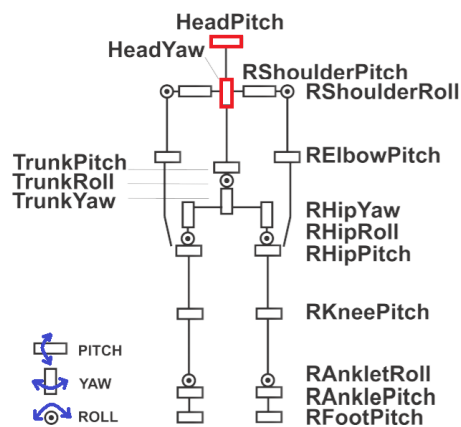
O modelo é iniciado no Experimento 1 (EXP 1) somente com os elementos em azul. No Experimento 2 (EXP 2) são adicionados os elementos em vermelho. No Experimento 3 (EXP 3) são adicionados os elementos em amarelo. A composição de cada estrutura, quanto aos mecanismos perceptivos, atencionais, de aprendizado e memórias presentes, foi inspirada nos aspectos do respectivo subestágio. Os experimentos objetivam explorar a habilidade de aprender como acompanhar, rastrear e procurar por outros agentes ou objetos. Estes experimentos são realizados em cenários com a agente cognitivo-atencional, humanoide Marta, em um ambiente com objetos de tamanhos, formas e cores diferentes, e um agente Pioneer 3DX, que possui o Algoritmo de Braitenberg, como ilustrado na Figura 18 (a), representando um corpo móvel no ambiente.

Cubos estáticos de tamanhos variados, nas cores azul (R: 0, G: 0, B: 255) e verde (R: 0, G: 255, B: 0) são distribuídos pelo ambiente de acordo com o propósito de cada experimento. Um robô Pioneer 3DX se movimenta pelo o ambiente e desvia dos cubos, simulando um corpo móvel. A humanoide Marta executa o algoritmo cognitivo-atencional do subestágio correspondente. Além dos atuadores físicos da agente (capazes de rotacionar os motores *HeadPitch* e *HeadYaw* para direções aleatórias), será considerado também um atuador virtual (interno ao agente) que será utilizado para simular o movimento dos olhos (movimentos sacádicos). Este atuador tipicamente selecionará um ponto do espaço visual para onde o agente direcionará o seu foco (fóvea), promovendo o aumento da resolução do sensor de visão em cada subestágio. A humanoide pode orientar sua fóvea, seus motores do pescoço e promover o aumento de sua acuidade visual, de acordo com cada nível de desenvolvimento. Os motores utilizados para movimentação da humanoide são os motores do pescoço (*HeadPitch* e *HeadYaw*), como ilustrado na Figura 18 (b). A agente pode orientar sua fóvea para uma das posições indicadas na Figura 18 (c).

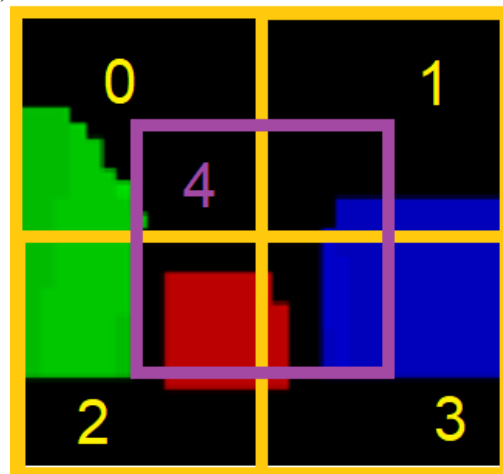
Os experimentos serão validados com base nos parâmetros e procedimentos descritos nos experimentos propostos por Berto (2020) (demonstrados no Anexo A).



(a)



(b)



(c)

Figura 18 – (a) humanoide Marta (algoritmo cognitivo-atencional), Robô Pioneer P3DX (algoritmo de Braitenberg) e objetos estáticos no Simulador Coppeliasim. (b) Motores utilizados (em **vermelho**) para os experimentos relacionados com a habilidade de aprender a rastrear e acompanhar objetos e outros agentes. Fonte: (TOMAZELA; COLOMBINI, 2019) (adaptado). (c) Posições para as quais a fôvea da agente pode ser posicionada.

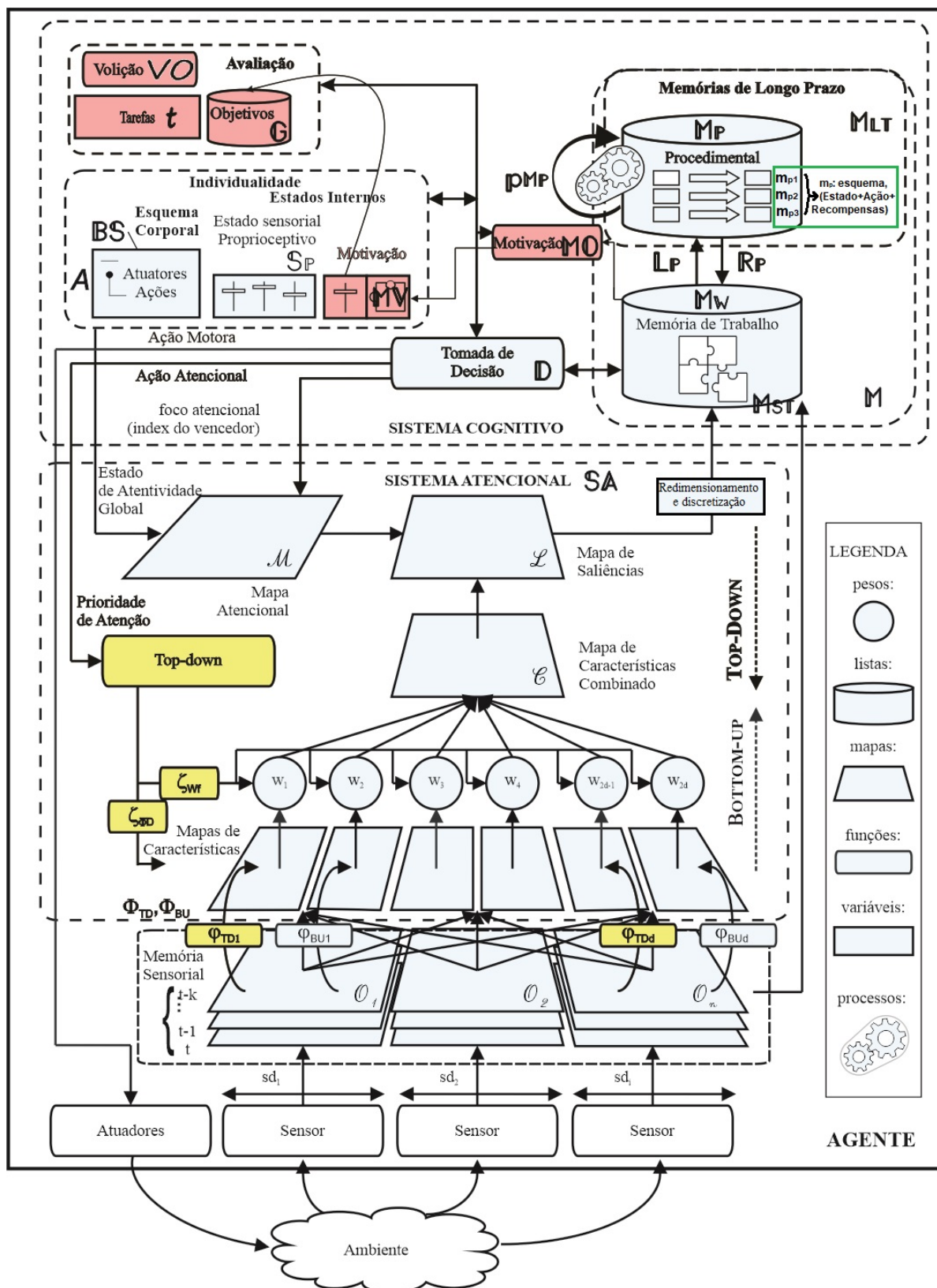


Figura 19 – Modelo interno da agente humanoide Marta. O modelo é iniciado no EXP 1 somente com os elementos em azul. No EXP 2 são adicionados os elementos em vermelho. No EXP 3 são adicionados os elementos em amarelo.

4.6.1 (EXP-01) 1º Subestágio: Uso de Reflexos

A estrutura do modelo do agente neste experimento é representada pelos elementos em azul da Figura 19. Tipicamente, nesse subestágio, os estados e ações serão inicializados com uma baixa discretização/resolução (em uma analogia à baixa capacidade de controle muscular da criança e sua baixa acuidade visual) que será gradualmente incrementada pelo algoritmo de aprendizado. Para simular a baixa acuidade visual neste subestágio, adotou-se inicialmente uma resolução para o sensor de visão de 64 x 64 pixels.

A criação automática de novos elementos na Memória Procedimental ($m_p \in M_P$) será interpretada no contexto do presente trabalho como a criação de um novo objeto de memória no CST (com uma nova tabela de estados-ações com algum nível de discretização para os estados e ações) associado a um *codelet* que executa o procedimento presente no objeto de memória. A agente inicia sua operação tipicamente com um único elemento (m_p) na Memória Procedimental (M_P) com essa tabela populada aleatoriamente, podendo eventualmente alguns pares estado-ação (tipicamente associados a condições de sobrevivência da agente no mundo) ter um preenchimento particular *a priori*. Essa condição de aptidão *inata* para algumas tarefas (*bootstrap condition*) encontra análogo, por exemplo, na ação de sugar de mamíferos, e não será objeto de estudo no presente trabalho.

Os estímulos visuais recebidos pela agente neste subestágio serão tratados pelo agente como estímulos *bottom-up*. As características *bottom-up* são combinadas em um mapa de características combinado que, com o mapa atencional, formam o mapa de saliências. A seleção da característica mais relevante no mapa de saliências é implementada através de um algoritmo *winner-takes-all*. O mapa de saliência L resultante, o vencedor (*winner*) do processo competitivo e suas observações são armazenados na Memória de Trabalho M_W . O sistema decisório (D) da robô decidirá entre executar movimentos aleatórios dos atuadores ou executar o procedimento m_p presente na Memória de Trabalho (M_W). A taxa de exploração será tipicamente muito alta no início da fase, sendo gradativamente reduzida com o decorrer do aprendizado da agente. O agente será modelado com Memória Sensorial, Memória de Trabalho (de curto-prazo) e Memória Procedimental (de longo-prazo). Nenhum modelo de intencionalidade ou motivação será adotado. A resposta da agente será investigada para a presença de objetos com diferentes cores, velocidades e posições em seu campo visual conforme os experimentos do subestágio previstos em Berto (2020).

Esse experimento propõe ainda a existência de um processo computacional a partir do qual a formação de **reações reflexas** pode ser modelada. Ou seja, propõe-se um processo computacional capaz de relacionar os sistemas sensório e motor promovendo o aprendizado procedimental de relações espaço-temporais de causa-efeito. Um *cognit* de Aprendizado Procedimental (L_P) que atua sobre a Memória de Trabalho (M_W) será utilizado para tal. Esse *cognit* (L_P), que fará uso de Aprendizado por Reforço (RL), será tipicamente disparado após a realização de ações pela agente, refinando seu aprendizado sobre o procedimento conhecido, ou seja, atualizando o conteúdo do $m_p \in M_P$. Particularmente, um reforço fortemente positivo será conferido à agente sempre que ocorrer o sincronismo espacial e temporal entre eventos sensórios e motores, estimulando a associação de tais eventos. Esse aprendizado será tipicamente disparado, por exemplo, quando a Memória de Trabalho (M_W) contiver um vencedor (*winner*) do processo competitivo que esteja no ponto central do processamento visual, com vistas a estabelecer uma nova relação reflexiva a um estímulo visual. A seguir destacamos todos os elementos cognitivos presentes neste experimento.

- **Sensores:** visão;
- **Features *bottom-up*:** $\mathcal{F}_1$, $\mathcal{F}_2$, $\mathcal{F}_3$ e $\mathcal{F}_4$;
- **Memórias:**
 - **Curto-prazo:** Memória Sensorial e Memória de Trabalho;
 - **Longo-prazo:** Memória Procedimental.

- **Ações motoras:**

- a_{M1} : Manter o foco (atuadores físicos e virtual mantêm suas posições);
- a_{M2} : Rotacionar o atuador *NeckYaw* 0,1 radiano para a esquerda;
- a_{M3} : Rotacionar o atuador *NeckYaw* 0,1 radiano para a direita;
- a_{M4} : Rotacionar o atuador *HeadPitch* 0,1 radiano para cima;
- a_{M5} : Mover o atuador *HeadPitch* 0,1 radiano para baixo;
- a_{M6} : Posicionar o atuador virtual para a posição 0 da Figura 18 (c);
- a_{M7} : Posicionar o atuador virtual para a posição 1 da Figura 18 (c);
- a_{M8} : Posicionar o atuador virtual para a posição 2 da Figura 18 (c);
- a_{M9} : Posicionar o atuador virtual para a posição 3 da Figura 18 (c);
- a_{M10} : Posicionar o atuador virtual para a posição 4 da Figura 18 (c).

- **Aprendizado Procedimental L_P :** implementado através do algoritmo de aprendizado por reforço Q-Learning;

- **Estados da agente:** $\mathcal{L}$;
- **Ações:** A ;
- **Discretização do espaço de estados:** 2 níveis para cada estado (2^{16} estados);
- **Taxa de exploração:** Inicia em 0,95 e decai linearmente para 0 na última execução;
- **Taxa de aprendizado:** 0,9;
- **Taxa de desconto temporal:** 0,99;
- **Número máximo de tentativas de execução (episódios):** 200;
- **Número máximo de passos em uma tentativa de execução:** 500;
- **Estrutura de reforço:**
 - * – $r = r + 1$ para cada novo dado inserido ao conceito sob investigação na memória procedimental;
 - * – $r = r + 1$ se a agente manter ou direcionar um atuador (atuadores motores *NeckYaw*, *HeadPitch* ou atuador virtual da fóvea) para a direção do vencedor do foco atencional;
 - * – $r = r - 10$ se a agente cair, ultrapassar os limites de seus atuadores motores ou se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações (500); (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

4.6.2 (EXP-02) 2º Subestágio: Reações Circulares Primárias

Como o modelo do agente proposto é incremental, a estrutura do modelo é representada pelos elementos em azul e em vermelho da Figura 19. Os blocos de memórias e mecanismos de aprendizado são mantidos os mesmos e a agente utiliza a QTable do estágio anterior em seu Aprendizado Procedimental. Espera-se que as reações reflexas, que tiveram início no 1º subestágio, eventualmente comecem a mostrar seus efeitos com maior constância com o avanço do processo de aprendizado. Desta forma, especula-se que a simples redução gradativa da taxa de exploração e o

aumento da taxa de usufruto do algoritmo de RL levem a uma maior exibição desses comportamentos. O refinamento desse procedimento prossegue durante este novo subestágio. Ainda, de forma a modelar a ampliação da capacidade muscular e da acuidade visual da criança, propõe-se a realização de um refinamento dos pares estado-ação em m_p . Esse procedimento tem potencial para gerar a construção gradativa de esquemas sensório-motores mais elaborados.

Com a evolução das iterações de treinamento, espera-se que ocorra uma estabilização do aprendizado para alguns conjuntos de pares estado-ação. Especulamos que a estabilização do aprendizado possa ser um indicativo de que esses pares estão aptos a constituírem novos m_p específicos aptos para a realização de tarefas em particular, ou seja, capazes de conduzir o agente para estados específicos. O agente seria, portanto, capaz de coleccionar esses novos procedimentos cujo efeito é conhecido, podendo passar a utilizá-los mediante procedimento de Recordação Procedimental (R_P), que poderia ser desencadeada por reflexo ou aleatoriamente. A resposta da agente aos experimentos do 2º subestágio propostos por Berto (2020) será investigada.

Neste experimento, propõe-se um processo computacional a partir do qual a formação de **reações circulares primárias** poderia ser modelada, ou seja, propõe-se um processo computacional capaz de relacionar os sistemas sensório e motor do subestágio atual promovendo o aprendizado procedimental de relações espaço-temporais de causa-efeito.

Do ponto de vista motor, a agente é modelada como possuindo *cognits* inatos capazes de posicionar seu atuador virtual e rotacionar os atuadores *HeadPitch* e *HeadYaw*, a exemplo do que ocorreu no EXP-01. Do ponto de vista sensório, também a exemplo do que ocorreu no EXP-01, os estímulos recebidos são tratados pela agente como estímulos *bottom-up*. As características *bottom-up* são combinadas em um mapa de características combinado que, com o mapa atencional, forma o mapa de saliências. A seleção da característica mais relevante no mapa de saliências é implementada através de um algoritmo *winner-takes-all*. O mapa de saliência L resultante, o vencedor (*winner*) do processo competitivo e suas observações são armazenados na Memória de Trabalho M_W .

O novo elemento inserido na arquitetura é um *cognit* de Aprendizado Procedimental (L_P) que atua sobre a Memória de Trabalho (M_W) e que busca modelar a formação e atualização de reações circulares primárias. Esse *cognit* de aprendizado (L_P) será tipicamente disparado em duas situações:

1. Sempre que ocorrer o sincronismo espacial e temporal entre eventos sensórios e motores, criando, nesse caso, um novo elemento na Memória Procedimental ($m_p \in M_P$) que associa tais eventos, ou
2. Sempre que um procedimento da Memória Procedimental ($m_p \in M_P$) for executado, nesse caso, refinando o aprendizado da agente sobre esse procedimento, ou seja, atualizando o conteúdo de um $m_p \in M_P$.

Esse aprendizado será disparado, por exemplo, quando a Memória de Trabalho (M_W) contiver um vencedor (*winner*) do processo competitivo que esteja no ponto central do processamento visual, ou seja, no ponto para onde a cabeça da agente estiver direcionada, com vistas a estabelecer uma nova relação reflexiva a um estímulo visual. Considerando que, no início, nenhum procedimento que relacione esses dois eventos concomitantes existirá na Memória Procedimental (M_P), um novo *cognit* que modela esse elemento (m_p) será criado. Já caso esse elemento (m_p) que modela a função reflexiva exista previamente na Memória Procedimental (M_P), sua atualização poderá ser realizada. Essa função em (m_p): i) passará a ser executada sempre que tal estímulo sensório ocorrer; ii) será modelada computacionalmente através de tabelas estado-ação, e iii) sua atualização será realizada utilizando Aprendizado por Reforço (RL).

- **Sensores:** visão;
- **Features bottom-up:** $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ e $\mathcal{F}_4$;
- **Memórias:**
 - **Curto-prazo:** Memória Sensorial e Memória de Trabalho;

- **Longo-prazo:** Memória Procedimental, com m_p do subestágio anterior.
- **Ações motoras:**
 - a_{M1} : Manter o foco;
 - a_{M2} : Mover o atuador *NeckYaw* 0,05 radiano para a esquerda;
 - a_{M3} : Mover o atuador *NeckYaw* 0,05 radiano para a direita;
 - a_{M4} : Mover o atuador *HeadPitch* 0,05 radiano para cima;
 - a_{M5} : Mover o atuador *HeadPitch* 0,05 radiano para baixo;
 - a_{M6} : Posicionar o atuador virtual para a posição 0 da Figura 18 (c);
 - a_{M7} : Posicionar o atuador virtual para a posição 1 da Figura 18 (c);
 - a_{M8} : Posicionar o atuador virtual para a posição 2 da Figura 18 (c);
 - a_{M9} : Posicionar o atuador virtual para a posição 3 da Figura 18 (c);
 - a_{M10} : Posicionar o atuador virtual para a posição 4 da Figura 18 (c).
- **Avaliação:** Lista de Objetivos G e Tarefas t inicialmente vazias;
- **Motivação M_V :** Inicialmente vazia;
- **Função de Motivação MO :** Todo esquema da Memória de Trabalho (M_W) que não puder ser localizado na Memória Procedimental da agente M_P adiciona um elemento em M_V com um nível de intensidade que se altera com o tempo. Para cada elemento em M_V , o nível de intensidade aumenta a cada novo estado caso o esquema continue não sendo localizado na Memória Procedimental M_P e diminui a cada novo estado com a localização do esquema na Memória Procedimental M_P .
- **Volição:** Para cada elemento m_v em M_V :
 - se o nível de intensidade do elemento m_v for fortemente positivo: objetivo de adquirir o esquema que originou a adição do elemento à m_v . Objetivo de executar a ação relacionada ao esquema. Diminuir o nível de intensidade do elemento m_v ;
 - se o nível de intensidade do elemento m_v for fortemente negativo: objetivo de explorar o ambiente. Aplicar o algoritmo de aprendizado e executar a ação escolhida a dentre as ações: a_{M2-5} . Aumentar o nível de intensidade do elemento m_v ;
 - senão: curso de Tomada de Decisão.
- **Recordação Procedimental R_P :** uma busca completa na Memória Procedimental é executada buscando por procedimentos que possam ser aplicados na situação corrente;
- **Aprendizado Procedimental L_P :** implementado através do algoritmo de aprendizado por reforço Q-Learning;
 - **Estados da agente:** $\mathcal{S}$;
 - **Ações:** A ;
 - **Discretização do espaço de estados:** 2 níveis para cada estado (2^{16} estados);
 - **Taxa de exploração:** Inicia em 0,95 e decai linearmente para 0 na última execução;
 - **Taxa de aprendizado:** 0,9;

- **Taxa de desconto temporal:** 0,99;
- **Número máximo de tentativas de execução (episódios):** 200;
- **Número máximo de passos em uma tentativa de execução:** 500.
- **Estrutura de reforço:**
 - * – $r = r + 1$ para cada novo dado inserido ao conceito sob investigação na memória procedimental;
 - * – $r = r + 1$ se a agente manter ou direcionar um atuador (atuadores motores *NeckYaw*, *HeadPitch* ou atuador virtual da fôvea) para a direção do vencedor (*winner*) do foco atencional;
 - * – $r = r - 10$ se a agente cair, ultrapassar os limites de seus atuadores motores (*NeckYaw* ou *HeadPitch*) ou se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

• **Tomada de Decisão D :**

- SE tarefa t e lista de objetivos G estão ambos vazios, ENTÃO direcionar o robô para o estímulo mais saliente;
- SE tarefa t está vazia e lista de objetivos G não está vazia, ENTÃO remover o objetivo g de maior prioridade de G e fazer dele a nova tarefa t ;
- SE tarefa t requer um novo procedimento conhecido, ENTÃO verificar o estado s e executar a ação a correspondente;
- SE tarefa t requer um procedimento desconhecido, ENTÃO aplicar o algoritmo de aprendizado e executar a ação escolhida a ;
- SE existir um estímulo relacionado a um procedimento que esteja na Memória de Trabalho (W_M), executá-lo;
- Caso contrário, executar periodicamente ações motoras aleatórias;

O mapa de saliência L resultante, o vencedor (*winner*) do processo competitivo e suas observações são armazenados na Memória de Trabalho M_W . Esta atualização estimula recordações e aprendizado na Memória Procedimental. Como a memória está inicialmente vazia, nenhuma recordação é esperada nesse momento. Um dos mecanismos disponíveis que pode estimular a agente a completar o conceito é a motivação. Dessa forma, tão logo o primeiro novo dado atinja M_P , espera-se que uma motivação seja gerada para fomentar a investigação desse esquema. A função de volição transforma essa motivação em um objetivo, adicionando-a à lista de Objetivos da agente, que estava inicialmente vazia. A função de Tomada de Decisão D é responsável por transformar esse objetivo em uma tarefa, inserindo-a na Memória de Trabalho M_W de forma a recordar o procedimento para esta tarefa. Como a Memória Procedimental M_P não tem inicialmente procedimentos armazenados, pode decidir iniciar um novo Aprendizado Procedimental L_P para esta tarefa. A cada passo do aprendizado, os pares estado-ação são armazenados na memória procedimental como uma receita para a tarefa de investigação de um novo esquema.

A função de motivação foi modelada neste trabalho representando a curiosidade da agente. Neste subestágio, quando os elementos gerados por esta função possuírem valor nulo, a agente seguirá seu curso de Tomada de Decisão. Quando os elementos possuírem valores fortemente positivos, a agente terá como objetivo explorar os esquemas que ainda não possui em sua Memória Procedimental M_P , tendo como objetivo executar as ações relacionadas aos esquemas que promoveram a criação de elementos de Motivação m_v . Quando os elementos possuírem valores fortemente negativos, a agente terá como objetivo adquirir novos estados, escolhendo dentre as ações de movimentação (a_{M2-5}).

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações (500); (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

4.6.3 (EXP-03) 3º Subestágio: Reações Circulares Secundárias

Neste experimento, a estrutura do modelo é representada por todos os elementos da Figura 19. Os blocos de memórias e mecanismos de aprendizado são mantidos os mesmos e a agente continua a aprender sobre a mesma QTable. Nesse subestágio amplia-se o refinamento de estados e ações que pode levar ao surgimento de novos esquemas de complexidade crescente. O mecanismo *top-down* da atenção passa a ser empregado nesta fase. A intencionalidade da agente será simulada estabelecendo-se objetivos específicos, que poderão ser atendidos mediante *recalls* de procedimentos m_p presentes na memória M_P que sejam capazes de atender às expectativas da agente. Como o agente computa sobre possíveis estados e ações em seu processo de aprendizado, os objetivos são definidos como estados desejados do futuro de acordo com o estado presente. Para objetivos complexos, os procedimentos m_p presentes na memória M_P podem ser combinados em novos procedimentos, de forma a se adaptar ao ambiente. Passa-se a ter a coordenação de procedimentos para atingir o objetivo da agente. Desta forma, reações circulares secundárias poderão ser estabelecidas. A resposta da agente aos experimentos propostos por Berto (2020) será investigada.

Neste experimento, propõe-se um processo computacional a partir do qual a formação de **reações circulares secundárias** poderia ser modelada, ou seja, propõe-se um processo computacional onde o agente é capaz de desenvolver a intencionalidade para executar uma ação, ou, em outros termos, selecionar intencionalmente uma ação atômica que permita atingir um objetivo.

Assim como no subestágio anterior, a robô utiliza das saliências formadas mecanismo atencional em conjunto com o desenvolvimento de reações circulares a partir de seus *cognits* inatos por sua Memória Procedimental e Aprendizado Procedimental. Porém, neste caso, o foco da robô torna-se externo e a mesma age intencionalmente. A robô implementa os quatro mapas de características *bottom-up* e três mapas de características *top-down*. Os estímulos endógenos e exógenos competem pelo foco da atenção. Nos casos em que ambos despertem interesse por uma mesma característica, os efeitos são combinados em um mapa de características combinado, potencializando a chance de focar a atenção naquele elemento. O mapa de características combinado, com o mapa atencional, forma o mapa de saliências. A seleção da característica mais relevante no mapa de saliências é implementada através de um algoritmo *winner-takes-all*. O sistema decisório da robô tem influência do módulo de tomada de decisão, que rotaciona os atuadores *HeadPitch* e *HeadYaw* da robô em direção ao estímulo mais saliente e do desenvolvimento das reações circulares pela Memória Procedimental e Aprendizado Procedimental. A robô possui Memória Sensorial e Memória de Trabalho (de curto-prazo) e Memória Procedimental (de longo-prazo).

- **Sensores:** visão;
- **Features *bottom-up*:** $\mathcal{F}_1, \mathcal{F}_2, \mathcal{F}_3$ e $\mathcal{F}_4$;
- **Features *top-down*:** $\mathcal{F}_5, \mathcal{F}_6$ e $\mathcal{F}_7$;
- **Memórias:**
 - **Curto-prazo:** Memória Sensorial e Memória de Trabalho;
 - **Longo-prazo:** Memória Procedimental, inicialmente vazia.
- **Ações motoras:**
 - a_{M1} : Manter o foco (atuadores físicos e virtual mantém suas posições);
 - a_{M2} : Rotacionar o atuador *NeckYaw* 0,01 radiano para a esquerda;
 - a_{M3} : Rotacionar o atuador *NeckYaw* 0,01 radiano para a direita;
 - a_{M4} : Rotacionar o atuador *HeadPitch* 0,01 radiano para cima;

- a_{M5} : Mover o atuador *HeadPitch* 0,01 radiano para baixo;
 - a_{M6} : Posicionar o atuador virtual para a posição 0 da Figura 18 (c);
 - a_{M7} : Posicionar o atuador virtual para a posição 1 da Figura 18 (c);
 - a_{M8} : Posicionar o atuador virtual para a posição 2 da Figura 18 (c);
 - a_{M9} : Posicionar o atuador virtual para a posição 3 da Figura 18 (c);
 - a_{M10} : Posicionar o atuador virtual para a posição 4 da Figura 18 (c);
 - a_{M11} : Rotacionar o atuador *NeckYaw* 0,01 radiano na direção da região do foco atencional;
 - a_{M12} : Rotacionar o atuador *HeadPitch* 0,01 radiano na direção da região do foco atencional;
 - a_{M13} : Rotacionar o atuador *NeckYaw* -0,01 radiano na direção da da região do foco atencional;
 - a_{M14} : Rotacionar o atuador *HeadPitch* -0,01 radiano na direção da região do foco atencional.
- **Ações atencionais:**
 - a_{A1} : Direcionar o foco atencional considerando a região em que uma determinada cor desejada é destacada de acordo com a característica *top-down* $\mathcal{F}_5$;
 - a_{A2} : Direcionar o foco atencional considerando a região em que uma determinada distância desejada é destacada de acordo com a característica *top-down* $\mathcal{F}_6$;
 - a_{A3} : Direcionar o foco atencional considerando a região destacada de acordo com a característica *top-down* $\mathcal{F}_7$.
 - **Avaliação:** Lista de Objetivos G e Tarefas t inicialmente vazias;
 - **Motivação MV :** Inicialmente vazia;
 - **Função de Motivação MO :** Todo esquema da Memória de Trabalho (M_W) que não puder ser localizado na Memória Procedimental da agente M_P adiciona um elemento em MV com um nível de intensidade que se altera com o tempo. Para cada elemento em MV , o nível de intensidade aumenta a cada novo estado caso o esquema continue não sendo localizado na Memória Procedimental M_P e diminui a cada novo estado com a localização do esquema na Memória Procedimental M_P .
 - **Volição:** Para cada elemento m_v em MV :
 - se o nível de intensidade do elemento m_v for fortemente positivo: objetivo de adquirir o esquema que originou a adição do elemento à m_v . Objetivo de executar a ação relacionada ao esquema. Diminuir o nível de intensidade do elemento m_v ;
 - se o nível de intensidade do elemento m_v for fortemente negativo: objetivo de explorar o ambiente. Aplicar o algoritmo de aprendizado e executar a ação escolhida a dentre as ações: a_{A1-3} . Aumentar o nível de intensidade do elemento m_v ;
 - senão: curso de Tomada de Decisão.
 - **Recordação Procedimental R_P :** uma busca completa na Memória Procedimental é executada buscando por tarefas que possam ser aplicadas na situação corrente;
 - **Aprendizado Procedimental L_P :** implementado através do algoritmo de aprendizado por reforço Q-Learning;
 - **Estados da agente:** $\mathcal{L}$;

- **Ações:** A ;
- **Discretização do espaço de estados:** 2 níveis para cada estado (2^{16} estados);
- **Taxa de exploração:** Inicia em 0,95 e decai linearmente para 0 na última execução;
- **Taxa de aprendizado:** 0,9;
- **Taxa de desconto temporal:** 0,99;
- **Número máximo de tentativas de execução:** 200;
- **Número máximo de passos em uma tentativa de execução (episódios):** 500;
- **Estrutura de reforço:**
 - * $-r = r + 1$ para cada novo dado inserido ao conceito sob investigação na memória procedimental;
 - * $-r = r + 1$ se a agente direcionar um atuador para a direção do vencedor do foco atencional;
 - * $-r = r + 1$ se a agente identificar a região em que uma determinada cor desejada é destacada de acordo com a característica *top-down* $\mathcal{F}_5$;
 - * $-r = r + 1$ se a agente identificar a região em que uma determinada distância desejada é destacada de acordo com a característica *top-down* $\mathcal{F}_6$;
 - * $-r = r + 1$ se a agente identificar a região destacada de acordo com a característica *top-down* $\mathcal{F}_7$;
 - * $-r = r - 10$ se a agente cair, ultrapassar os limites de seus atuadores ou se não encontrar saliências por 5 iterações seguidas.

• **Tomada de Decisão D :**

- SE tarefa t e lista de objetivos G estão ambos vazios, ENTÃO direcionar o robô para o estímulo mais saliente;
- SE tarefa t está vazia e lista de objetivos G não está vazia, ENTÃO remover o objetivo g de maior prioridade de G e fazer dele a nova tarefa t ;
- SE tarefa t requer um novo procedimento conhecido, ENTÃO verificar o estado s e executar a ação a correspondente;
- SE tarefa t requer um procedimento desconhecido, ENTÃO aplicar o algoritmo de aprendizado e executar a ação escolhida a ;
- SE existir um estímulo relacionado a um procedimento que esteja na Memória de Trabalho (M_W), executá-lo;
- Caso contrário, executar periodicamente ações motoras aleatórias;

O mapa de saliência L resultante, o vencedor (*winner*) do processo competitivo e suas observações são armazenados na Memória de Trabalho M_W . Esta atualização estimula recordações e aprendizado na Memória Procedimental. Um dos mecanismos disponíveis que pode estimular a agente a aprender o procedimento é a motivação. Dessa forma, tão logo o primeiro novo dado atinja M_P , espera-se que uma motivação seja gerada para fomentar a investigação desse esquema. A função de volição transforma essa motivação em um objetivo, adicionando-a à lista de Objetivos da agente, que estava inicialmente vazia. A função de Tomada de Decisão D é responsável por transformar esse objetivo em uma tarefa, inserindo-a na Memória de Trabalho M_W de forma a recordar o procedimento para esta tarefa. Como a Memória Procedimental M_P não tem inicialmente procedimentos armazenados, pode decidir iniciar um novo Aprendizado Procedimental L_P para esta tarefa. O processo é completado escolhendo-se ações adicionais $a_A \in A_A$

que possam enfatizar a atenção na área próxima às regiões destacadas através de ações *top-down*. A cada passo do aprendizado, os pares estado-ação são armazenados na Memória Procedimental como uma receita para a tarefa de investigação de um novo esquema.

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações (500); (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

5 RESULTADOS E DISCUSSÕES

"Uma teoria é algo que ninguém acredita, exceto quem a fez. Um experimento é algo que todos acreditam, exceto quem o fez."
(Albert Einstein)

Após modeladas as classes sensoriais, atencionais, de aprendizado e de validação, foram realizados experimentos para análise do processo de aprendizado cognitivo-atencional na humanoide Marta. Foram realizados três experimentos para validação e treinamento da arquitetura de integração, correspondentes aos três primeiros subestágios do período sensório-motor na Teoria de Jean Piaget (PIAGET, 1952). Os experimentos utilizam a humanoide Marta em comunicação com o sistema cognitivo-atencional. Nesta seção, discorre-se sobre:

- As observações do desenvolvimento do ciclo atencional, a elaboração dos mapas de características, atencionais e de saliências em cada subestágio;
- Os resultados obtidos com o algoritmo QLearning durante o treinamento do Aprendizado Procedimental em cada subestágio;
- As reações reflexas, reações circulares primárias e reações circulares secundárias identificadas em cada subestágio;
- O desempenho da agente quanto à avaliação proposta pelos experimentos de Berto (2020) em cada subestágio.

5.1 (EXP-01) 1º Subestágio: Uso de Reflexos

Neste subestágio é utilizada a humanoide Marta com um algoritmo cognitivo-atencional que possui somente os elementos em azul demonstrados na Figura 19. Nenhum modelo de intencionalidade ou motivação foi adotado, a robô possui neste subestágio apenas elementos de percepção *bottom-up*. Para simular a baixa acuidade visual neste subestágio, adotou-se uma resolução para o sensor de visão de 64 x 64 pixels. A resposta da agente foi investigada para a presença de objetos estáticos com diferentes cores e um robô Pioneer 3DX, que possui um algoritmo de Braitenberg implementado e se move com uma velocidade média de 0.05 m/s, se comportando como um corpo móvel no ambiente. O Aprendizado Procedimental foi executado durante 200 episódios. Ao fim de cada episódio, a recompensa adquirida e o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer 3DX é redefinida aleatoriamente. O experimento demonstra o curso atencional da seleção para percepção para uma agente executando apenas o curso *bottom-up* da atenção.

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações; (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

5.1.1 Validação do Sistema Atencional

A cena do simulador *CoppeliaSim* no instante $t=1s$ no 1º episódio é demonstrada na Figura 20 (a). A imagem obtida pelo sensor de visão no instante $t=1s$ no 1º episódio é demonstrada na Figura 20 (b). Os resultados do ciclo atencional para os estímulos obtidos nos instantes seguintes são demonstrados nas Figuras 20 (c-n).

Através da ferramenta de comunicação externa *RemoteAPI CoppeliaSim*, o sensor de visão da Humanoide Marta coleta informações sensoriais (intensidades de cores nos três canais RGB e distâncias relativas ao sensor de visão) do ambiente, formando objetos de memória que correspondem à estas observações, representados como $Color_{MO}$ e $Depth_{MO}$. Estes objetos são armazenados em quantidade limitada na memória sensorial pelos codelets *Color Buffer* e *Depth Buffer*, gerando os objetos de memória $ColorBuffer_{MO}$ e $DepthBuffer_{MO}$. Os níveis em RGB dos canais de cor vermelho, verde e azul para os pixels disponibilizados pelo sensor de visão no instante $t=1s$ do 1º episódio são demonstrados nas Figuras 20 (c), (d) e (e), respectivamente. As distâncias relativas (0m: muito próximo, 10m: muito distante) ao sensor de visão para o instante $t=1s$ do 1º episódio são demonstradas na Figura 20 (f). Os dados sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza.

Por meio dos objetos de memória criados $ColorBuffer_{MO}$ e $DepthBuffer_{MO}$, são construídos os mapas de características de atenção exógena (*bottom-up*) de intensidades de cores e de distância pelos codelets *Red bottom-up FeatureMap*, *Green bottom-up FeatureMap*, *Blue bottom-up FeatureMap* e *Distance bottom-up FeatureMap*, gerando mapas de características representados como objetos de memória: $RedFM_{BUMO}$, $GreenFM_{BUMO}$, $BlueFM_{BUMO}$ e $DistanceFM_{BUMO}$. Os mapas de características *bottom-up* para os canais de cor vermelho, verde e azul no instante $t=2s$ do 1º episódio são demonstrados nas Figuras 20 (g), (h) e (i). O mapa de características *bottom-up* para as distâncias relativas ao sensor de visão no instante $t=2s$ do 1º episódio são demonstradas na Figura 20 (j). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=2s$ do 1º episódio, em escala de cinza.

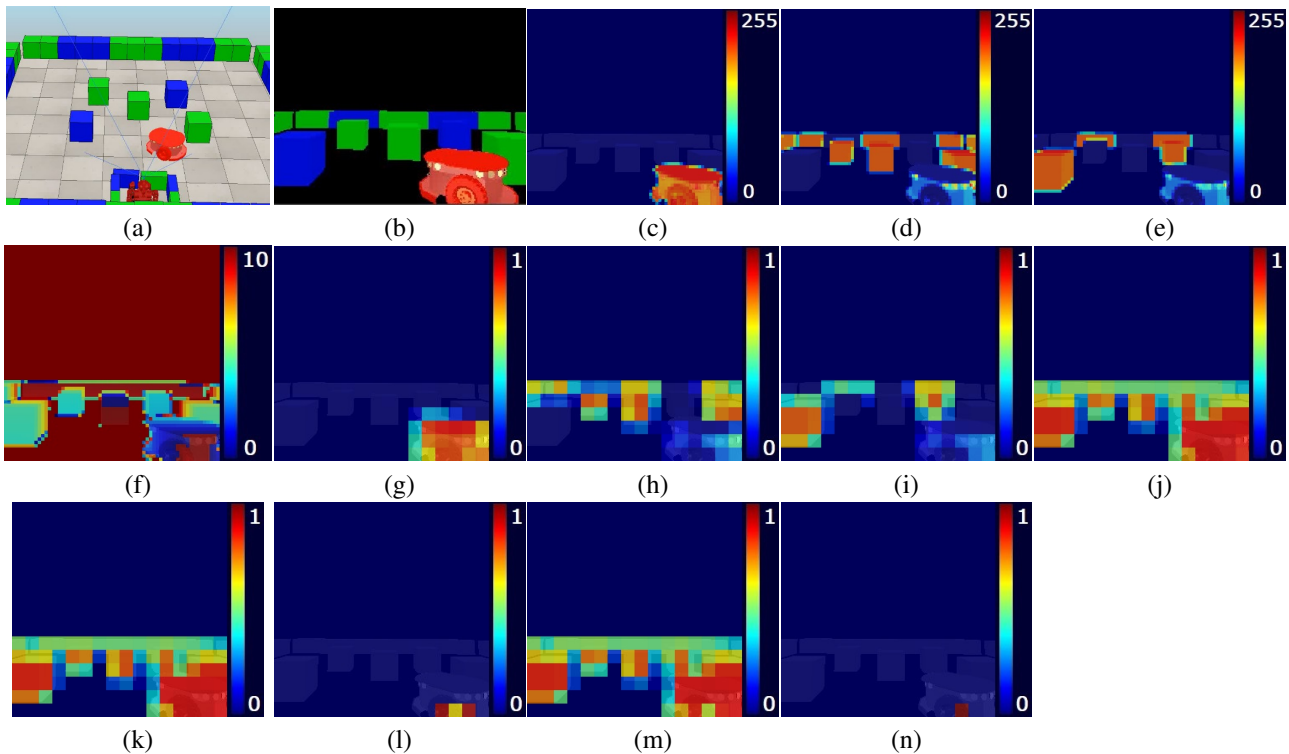


Figura 20 – (1º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental - (a) Cena no simulador *CoppeliaSim* ($t = 1s$); (b) Imagem do sensor de visão ($t = 1s$); (c) Valores do canal vermelho para o sensor de visão ($t = 1s$); (d) Valores do canal verde para o sensor de visão ($t = 1s$); (e) Valores do canal azul para o sensor de visão ($t = 1s$); (f) Distância relativa ao sensor de visão ($t = 1s$); (g) *Bottom-up Feature Map* para o canal vermelho ($t = 2s$); (h) *Bottom-up Feature Map* para o canal verde ($t = 2s$); (i) *Bottom-up Feature Map* para o canal azul ($t = 2s$); (j) *Bottom-up Feature Map* para distância ($t = 2s$); (k) Mapa de características combinado ($t = 3s$); (l) Mapa Atencional normalizado ($t = 3s$); (m) Mapa de Saliências ($t = 3s$); (n) Característica vencedora do ciclo atencional ($t = 3s$).

Os mapas de características $RedFM_{BUMO}$, $GreenFM_{BUMO}$, $BlueFM_{BUMO}$ e $DistanceFM_{BUMO}$ são combinados em um mapa de características combinado pelo codelet *CombinedFeatureMap*, conforme proposto pelo Modelo de Colombini (2016) (COLOMBINI, 2014), gerando o objeto de memória $CombFM_{MO}$. O Mapa Atencional é inicializado com valores unitários, simulando um ambiente sem vencedores, e é realimentado a cada iteração pelo codelet de Tomada de Decisão *Decision Making*. O mapa de saliências é obtido pelo codelet *SaliencyMap* através do produto entre o Mapa de Características Combinado ($CombFM_{MO}$) e o Mapa Atencional ($AttMap_{MO}$), também seguindo o Modelo de Colombini. O mapa de saliências $SalMap_{MO}$ influencia o codelet de Tomada de Decisão, definindo a característica vencedora de cada ciclo, que influencia no decorrer do ciclo atencional, com base nos ciclos inibitórios e excitatórios da operação *bottom-up*. O mapa de características combinado, o Mapa Atencional normalizado, o Mapa de Saliências e a característica vencedor do ciclo atencional no instante $t=3s$ do 1º episódio são demonstrados nas Figuras 20 (k), (l), (m) e (n). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza. Conforme demonstram as imagens, os mapas realçam as regiões sensoriais de interesse à agente.

5.1.2 Validação do Sistema de Aprendizado Procedimental

A Figura 21 apresenta os valores médios e os desvios padrão de recompensas obtidos pela agente e o número médio de ações a cada 10 episódios ao longo do Aprendizado Procedimental. A Figura 22 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo do Aprendizado Procedimental.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

Nos primeiros episódios, a agente explora os limites de seus atuadores devido à alta taxa de exploração, ao passo que promove o refinamento dos pares estado-ação para o atuador virtual de seleção da fôvea. Neste início, a ação a_{M1} (manter o foco) é pouco executada. Com o distanciamento do robô Pioneer 3DX, entre os episódios 48 e 51, o ciclo atencional direciona o foco de atenção para as regiões em que os estímulos *bottom-up* são mais fortes. O aumento na execução da ação a_{M4} (rotacionar *HeadPitch* para baixo) direciona o sensor de visão da agente para os blocos mais próximos que delimitam seu corpo e para seus próprios pés. Este comportamento estimula a exploração do corpo da agente, ocasionando o aumento da atuação do reflexo a_{M4} , que acarreta no posicionamento o atuador *HeadPitch* para fora dos limites de atuação. A agente é penalizada por ultrapassar o limite do atuador.

De forma a ilustrar o comportamento da agente com a exploração de seu corpo, são demonstrados os estímulos atencionais adquiridos nestas regiões nas Figuras 23 (a-o). Os estímulos e recompensas obtidos pela agente durante a exploração das regiões de seu próprio corpo reforçam os reflexos que ocasionaram estas posições. Nos episódios 60 e 140, mesmo com a aproximação do robô Pioneer 3DX, o foco de atenção da agente é mantido em seus pés, como demonstram as Figuras 23 (f-o). Os mapas sobrepõem a imagem obtida pelo sensor de visão em escala de cinza.

Entre os episódios 88 e 91, a execução de a_{M5} (rotacionar *HeadPitch* para cima) aumenta drasticamente, posicionando o sensor de visão para fora do cenário de simulação. A agente é penalizada por ultrapassar o limite do atuador e por não identificar saliências em 5 iterações consecutivas. A partir do episódio 180, com a reaproximação do robô Pioneer 3DX, a agente direciona o sensor de visão ao campo externo. No entanto, o foco atencional continua em regiões próximas da localização da agente, como ilustram as Figuras 23 (p-y). Os mapas sobrepõem as imagens obtidas pelo sensor de visão em escala de cinza.

Na Figura 23, são demonstrados:

- As cenas no *CoppeliaSim* nos instantes: (a) $t=4s$ do episódio 60, (f) $t=23s$ do episódio 60, (k) $t=63s$ do episódio 140, (p) $t=33s$ do episódio 180, (u) $t=53s$ do episódio 200.

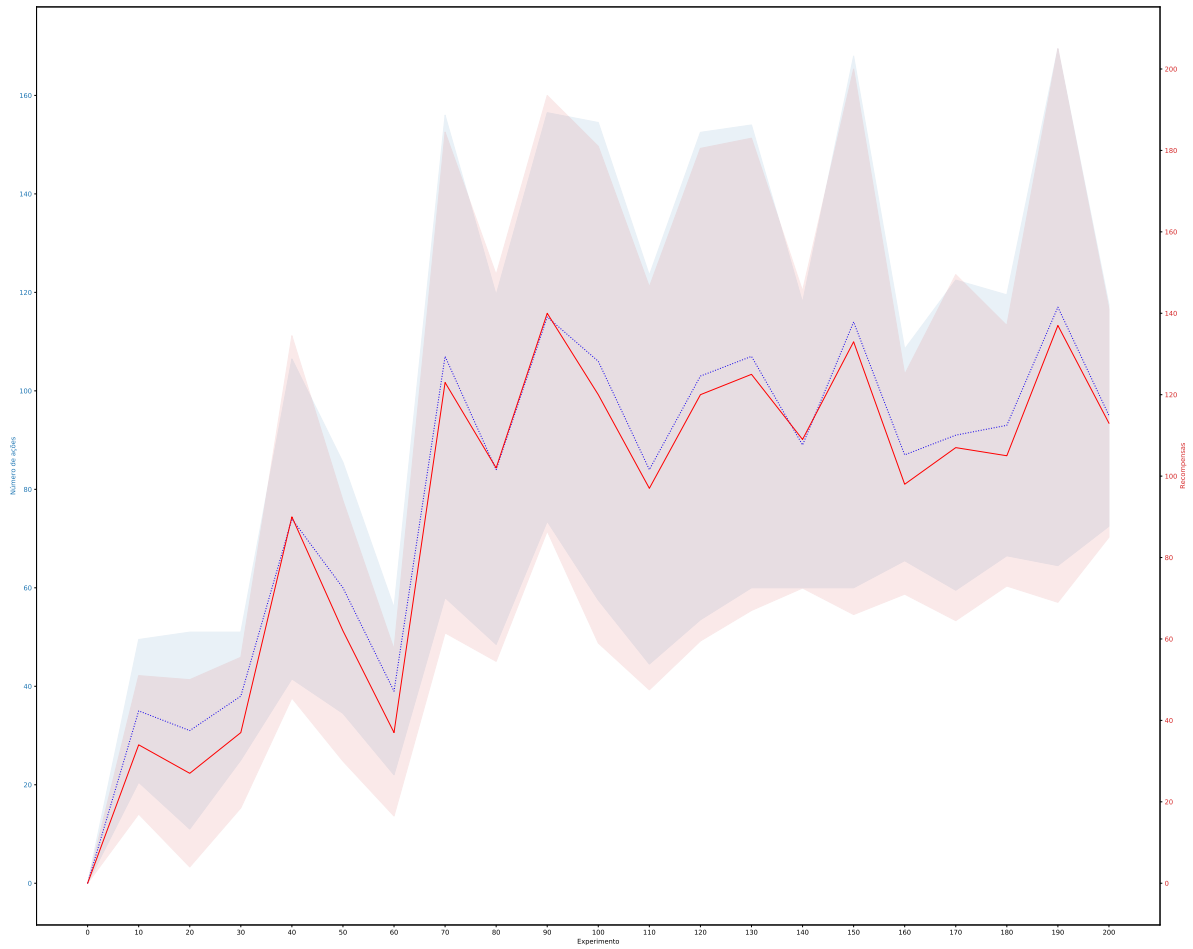


Figura 21 – (1º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.

- As imagens adquiridas pelo sensor de visão nos instantes: (b) $t=4s$ do episódio 60, (g) $t=23s$ do episódio 60, (l) $t=63s$ do episódio 140, (q) $t=33s$ do episódio 180, (v) $t=53s$ do episódio 200;
- Os Mapas Atencionais normalizados nos instantes: (c) $t=6s$ do episódio 60, (h) $t=25s$ do episódio 60, (m) $t=65s$ do episódio 140, (r) $t=35s$ do episódio 180, (w) $t=55s$ do episódio 200;
- Os Mapas de Saliências nos instantes: (d) $t=6s$ do episódio 60; (i) $t=25s$ do episódio 60, (n) $t=65s$ do episódio 140, (s) $t=35s$ do episódio 180, (x) $t=55s$ do episódio 200;
- As regiões vencedoras do ciclo atencional nos instantes: (e) $t=6s$ do episódio 60, (j) $t=25s$ do episódio 60, (o) $t=65s$ do episódio 140, (t) $t=35s$ do episódio 180, (y) $t=55s$ do episódio 200;

A partir do episódio 123, as execuções das ações a_{M6-10} (posicionamento da fôvea) passam a ser seguidas da execução da ação a_{M1} (manter o foco). Os estímulos recebidos reforçam a atuação dos reflexos em conjunto. A agente executa demais episódios com ao menos 15 ações executadas. Os valores máximos de ações executadas (162) e recompensa adquirida (200) são obtidos no último episódio. As médias do número de ações e recompensas adquiridas a cada 10 episódios apresentam crescimento ao longo dos episódios e se mantêm acima de 75 a partir do episódio 70. Ao final dos episódios do 1º subestágio, a tabela QTable construída pelo *Q-learning* representa a receita de execução das tarefas aprendidas e armazenadas.

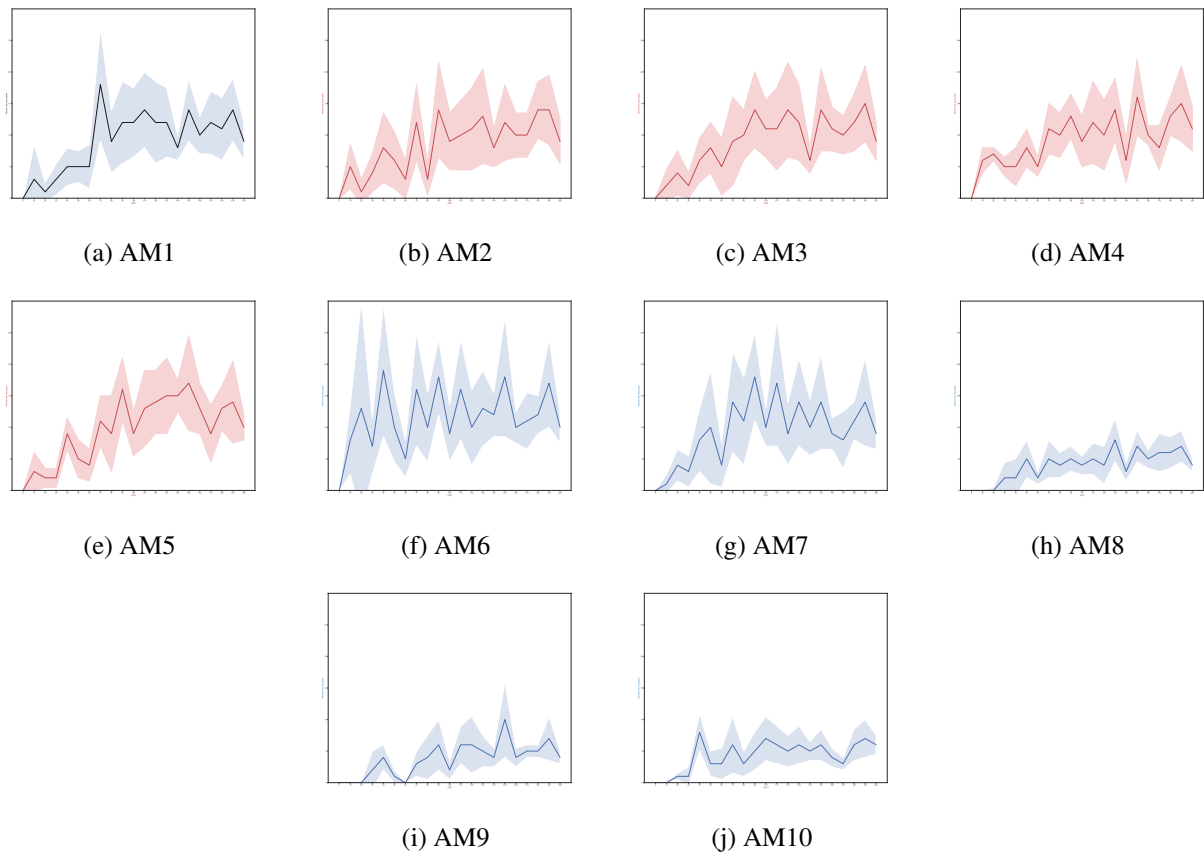


Figura 22 – (1º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios do Aprendizado Procedimental: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea.

Neste subestágio, a agente aprendeu a direcionar seus atuadores para as regiões próximas do seu corpo com o afastamento do robô Pioneer 3-DX. A agente também demonstrou a capacidade de relacionar a execução de ações de forma a aumentar sua recompensa adquirida, entre: (i-iv) as ações de movimentação a_{M2-5} e a ação de manter foco a_{M1} ; (v-vi) as ações de atualizar a posição da fóvea a_{M6-7} e a ação de manter foco a_{M1} . Os procedimentos poderão ser recordados nos próximos subestágios, bem como melhorados por novas ações da agente.

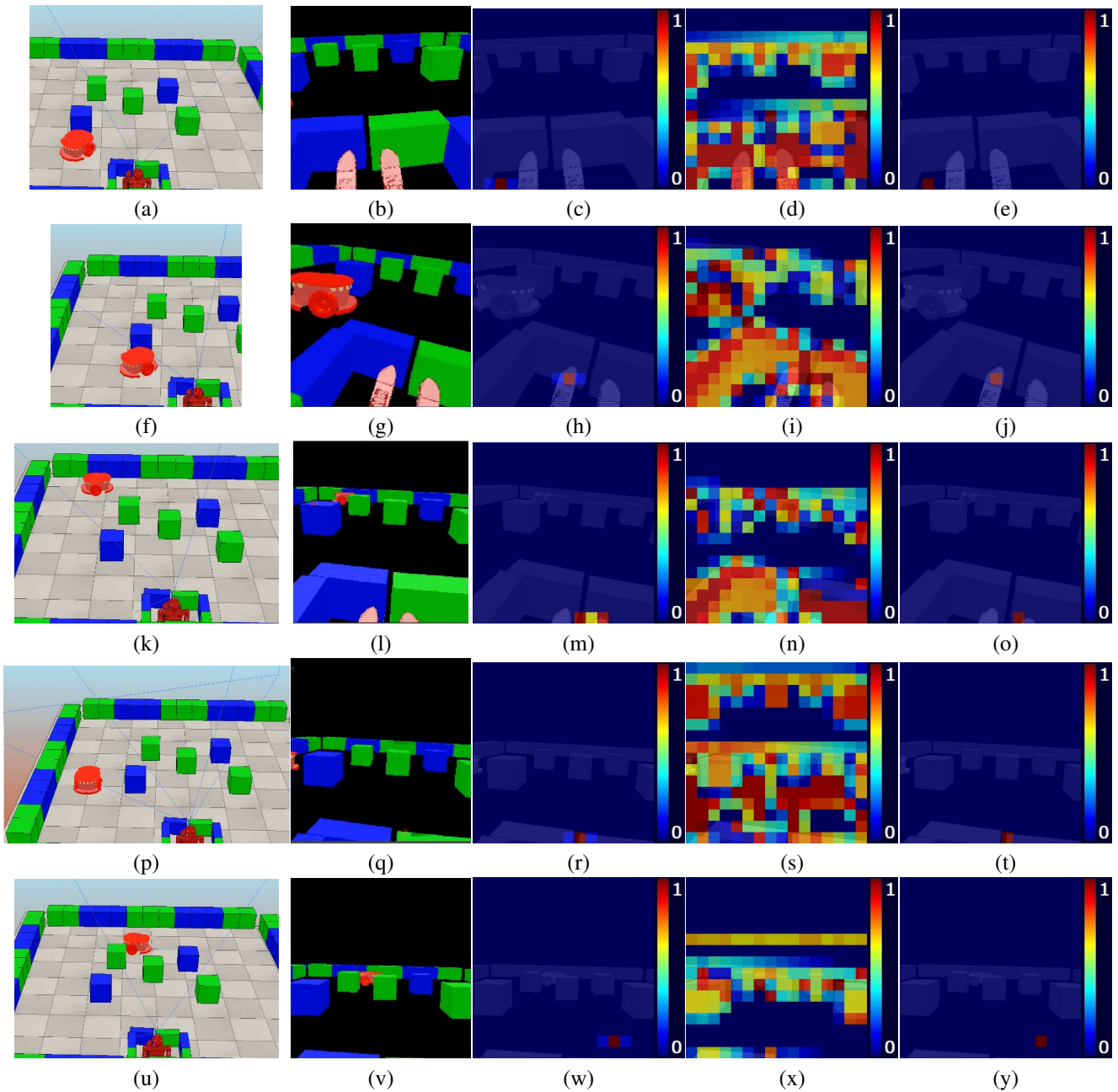


Figura 23 – (1º Subestágio) Dados sensoriais obtidos durante o Aprendizado Procedimental - (a) Cena do *CoppeliaSim* (Ep. 60, $t = 4s$); (b) Imagem do sensor de visão (Ep. 60, $t = 4s$); (c) Mapa Atencional normalizado (Ep. 60, $t = 6s$); (d) Mapa de Saliências (Ep. 60, $t = 6s$); (e) Característica vencedora do ciclo atencional (Ep. 60, $t = 6s$); (f) Cena do *CoppeliaSim* (Ep. 60, $t = 23s$); (g) Imagem do sensor de visão (Ep. 60, $t = 23s$); (h) Mapa Atencional normalizado (Ep. 60, $t = 25s$); (i) Mapa de Saliências (Ep. 60, $t = 25s$); (j) Característica vencedora do ciclo atencional (Ep. 60, $t = 25s$); (k) Cena do *CoppeliaSim* (Ep. 140, $t = 63s$); (l) Imagem do sensor de visão (Ep. 140, $t = 63s$); (m) Mapa Atencional normalizado (Ep. 140, $t = 65s$); (n) Mapa de Saliências (Ep. 140, $t = 65s$); (o) Característica vencedora do ciclo atencional (Ep. 140, $t = 65s$); (p) Cena do *CoppeliaSim* (Ep. 180, $t = 33s$); (q) Imagem do sensor de visão (Ep. 180, $t = 33s$); (r) Mapa Atencional normalizado (Ep. 180, $t = 35s$); (s) Mapa de Saliências (Ep. 180, $t = 35s$); (t) Característica vencedora do ciclo atencional (Ep. 180, $t = 35s$); (u) Cena do *CoppeliaSim* (Ep. 200, $t = 53s$); (v) Imagem do sensor de visão (Ep. 200, $t = 53s$); (w) Mapa Atencional normalizado (Ep. 200, $t = 55s$); (x) Mapa de Saliências (Ep. 200, $t = 55s$); (y) Característica vencedora do ciclo atencional (Ep. 200, $t = 55s$).

5.1.3 Validação do Aprendizado com os Experimentos de Berto

De forma a verificar o aprendizado da agente durante o 1º subestágio, foi utilizada a tabela QTable resultante do fim do último episódio do Aprendizado Procedimental para realizar os seguintes experimentos, adaptados dos experimentos cognitivos propostos por Berto (2020) para a avaliação de agentes cognitivos no 1º subestágio sensório-motor da Teoria de Jean Piaget (PIAGET, 1952) na tarefa de rastrear objetos. Os experimentos propostos por Berto (2020) são apresentados no Anexo A. Foram realizadas 100 execuções para cada experimento com, no máximo, 500 ações cada execução.

5.1.3.1 Experimento A - 1º Subestágio - Objeto em posição fixa e de cor primária

A robô Marta foi posicionada em frente, à 80cm do robô Pioneer 3DX, conforme ilustra a cena da Figura 24 (a). O robô Pioneer 3DX se manteve estático durante todas as execuções deste experimento. Ao fim de cada episódio, o número de ações executadas é zerado e os atuadores da humanoide Marta retornam para a posição inicial. A posição do robô Pioneer 3DX é mantida.

No 1º episódio, o sistema atencional da agente identifica o robô Pioneer 3DX como ponto mais saliente, conforme demonstram as Figuras 24 (c-e) para o instante $t = 30s$. No entanto, os ciclos inibitórios do processo *bottom-up* promovem a ascensão de outras regiões salientes: os blocos que circundam a agente, conforme demonstram as Figuras 24 (f-j); e o próprio corpo da agente, conforme demonstram as Figuras 24 (k-o). Os mapas sobrepõem a imagem obtida pelo sensor de visão em escala de cinza.

A Figura 25 apresenta os valores médios e os desvios padrão do número de ações executado a cada 10 episódios ao longo deste experimento. A Figura 26 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo deste experimento. O maior número de ações executadas (258) é obtido no episódio 97. O número médio de ações a cada 10 episódios se mantém acima de 100 desde o início deste experimento.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

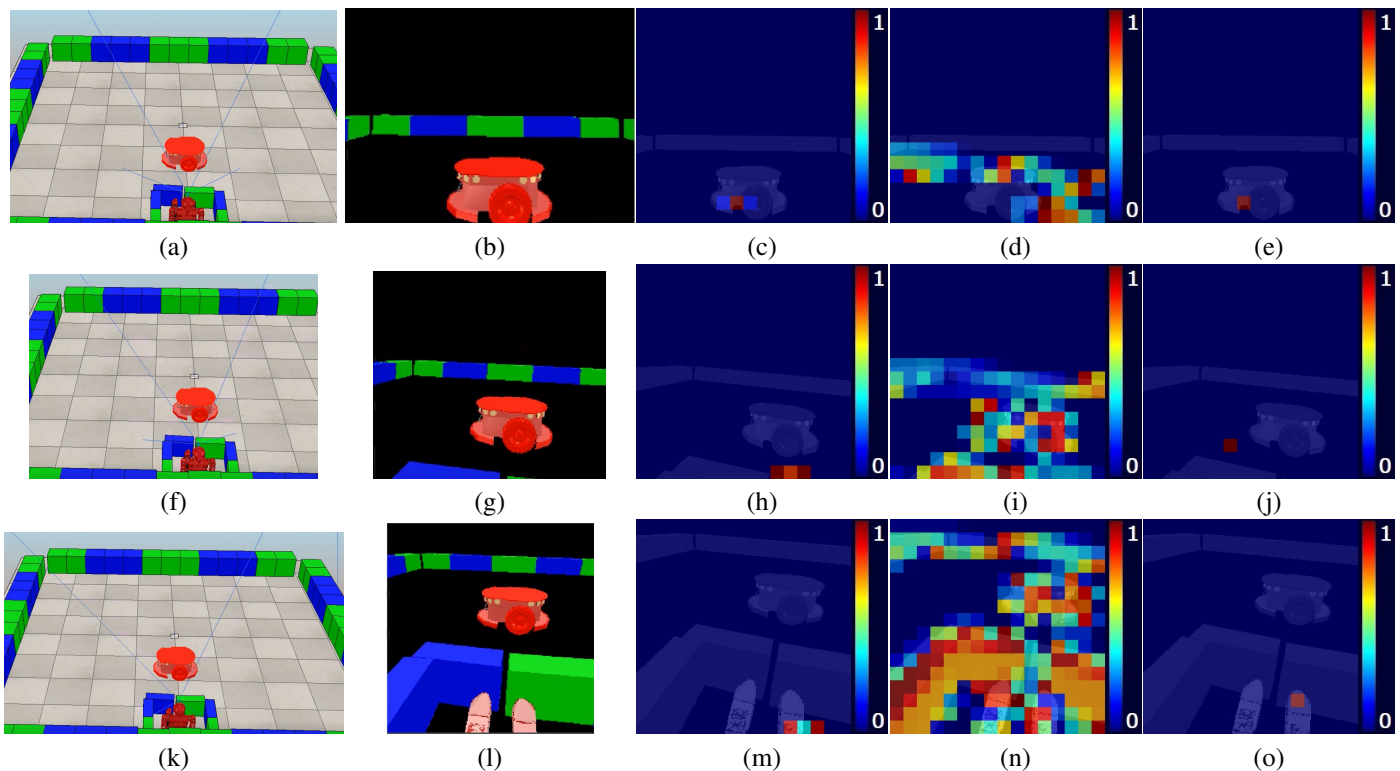


Figura 24 – (Berto 1º Subestágio - Exp. A) Dados sensoriais obtidos nas execuções do Exp. A. Objeto em posição fixa e de cor primária - (a) Cena do *CoppeliaSim* (Ep. 1, $t = 28s$). (b) Imagem do sensor de visão (Ep. 1, $t = 28s$); (c) Mapa Atencional normalizado (Ep. 1, $t = 30s$); (d) Mapa de Saliências (Ep. 1, $t = 30s$); (e) Característica vencedora do ciclo atencional (Ep. 1, $t = 30s$); (f) Cena do *CoppeliaSim* (Ep. 40, $t = 4s$). (g) Imagem do sensor de visão (Ep. 40, $t = 4s$); (h) Mapa Atencional normalizado (Ep. 40, $t = 6s$); (i) Mapa de Saliências (Ep. 40, $t = 6s$); (j) Característica vencedora do ciclo atencional (Ep. 40, $t = 6s$); (k) Cena do *CoppeliaSim* (Ep. 80, $t = 41s$). (l) Imagem do sensor de visão (Ep. 80, $t = 41s$); (m) Mapa Atencional normalizado (Ep. 80, $t = 43s$); (n) Mapa de Saliências (Ep. 80, $t = 43s$); (o) Característica vencedora do ciclo atencional (Ep. 80, $t = 43s$).

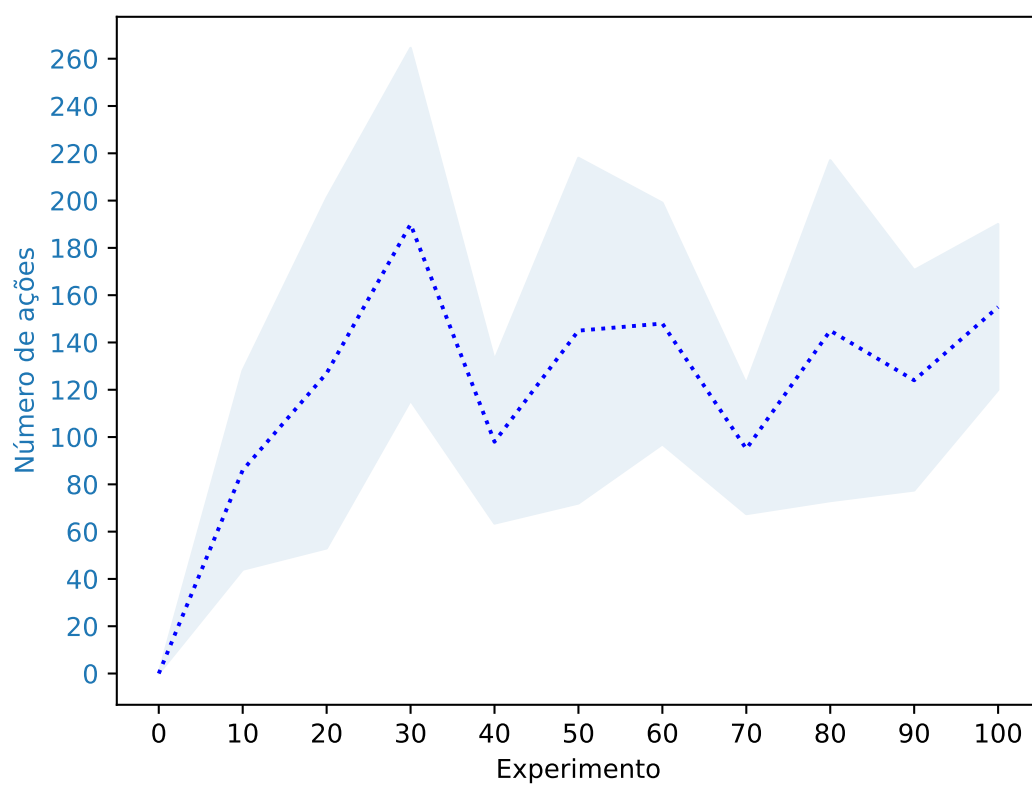


Figura 25 – (Berto 1º Subestágio - Exp. A) Médias e desvios padrão do número de ações executadas a cada 10 episódios.

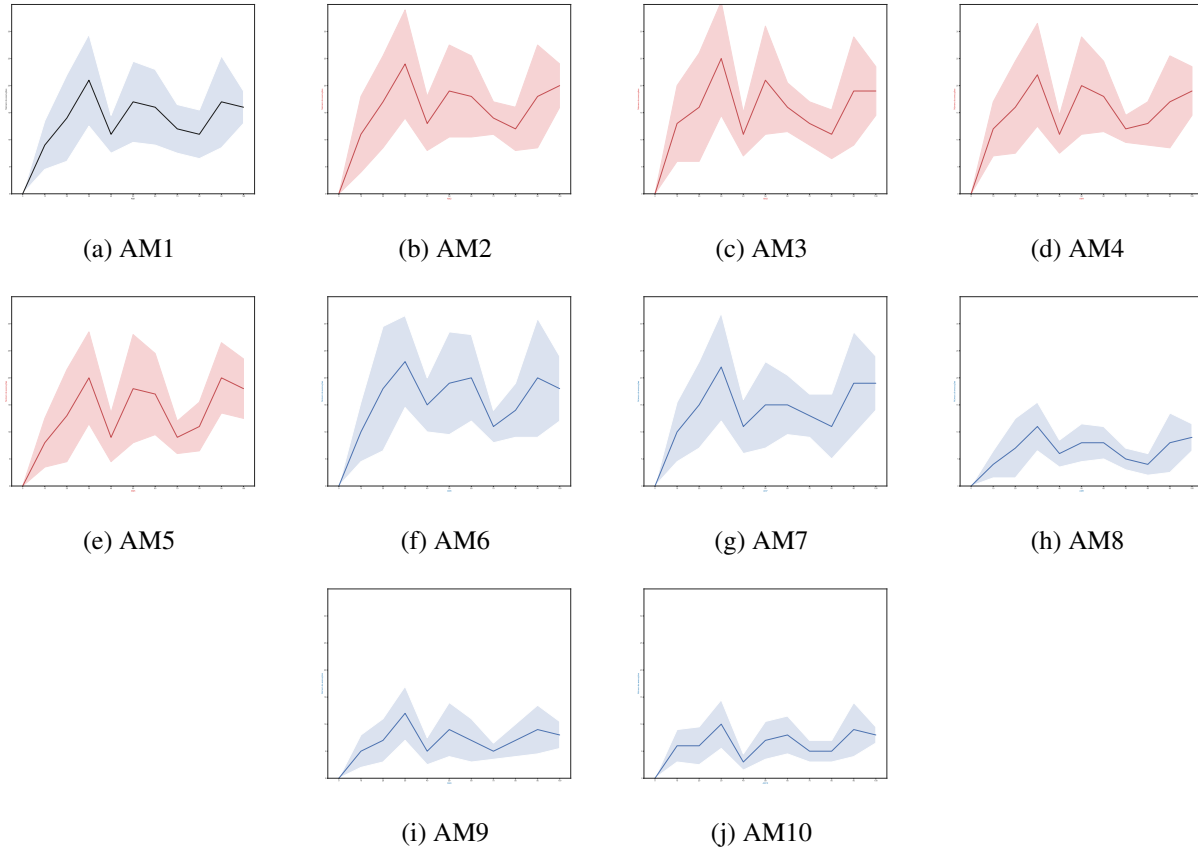


Figura 26 – (Berto - 1º Subestágio - Exp. A) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. (Berto 1º Subestágio - Exp. A) Dados sensoriais obtidos nas execuções do Exp. A. Objeto em posição fixa e de cor primária: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea.

5.1.3.2 Experimento B - 1º Subestágio - Objeto se movimentando lentamente e de cor primária

A robô Marta foi posicionada 80cm à frente do robô Pioneer P3DX, conforme ilustra a cena da Figura 27 (a). O robô Pioneer P3DX possui um algoritmo de Braitenberg e se movimenta com uma velocidade constante de 0.1m/s em todas as execuções deste experimento. Ao fim de cada episódio, o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer P3DX é redefinida aleatoriamente.

O sistema atencional da agente identifica o robô Pioneer P3DX como ponto mais saliente, mesmo em movimento, conforme demonstram as Figuras 27 (c-e) para o instante $t = 83s$ do episódio 60. No episódio 80, o distanciamento do robô Pioneer P3DX promove a ascensão das regiões salientes dos blocos que circundam a agente e os pés da agente, conforme demonstram as Figuras 27 (f-j). Os mapas sobrepõem a imagem obtida pelo sensor de visão em escala de cinza.

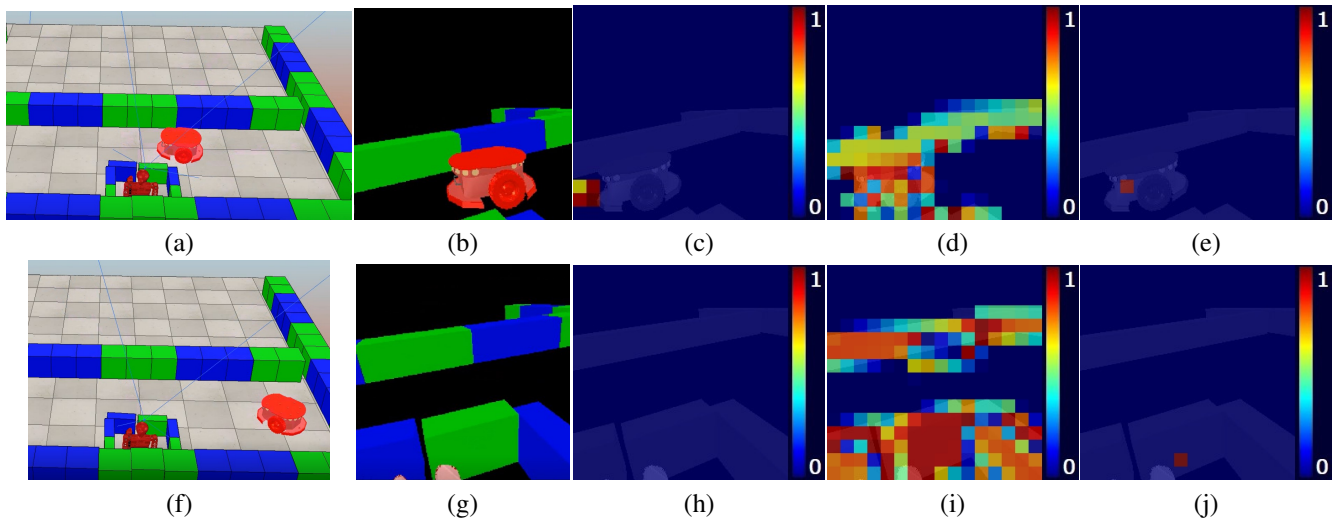


Figura 27 – (Berto 1º Subestágio - Exp. B) Dados sensoriais obtidos nas execuções do Exp. B. Objeto se movimentando lentamente e de cor primária - (a) Cena do *CoppeliaSim* (Ep. 60, $t = 81s$). (b) Imagem do sensor de visão (Ep. 60, $t = 81s$); (c) Mapa Atencional normalizado (Ep. 60, $t = 83s$); (d) Mapa de Saliências (Ep. 60, $t = 83s$); (e) Característica vencedora do ciclo atencional (Ep. 60, $t = 83s$); (f) Cena do *CoppeliaSim* (Ep. 80, $t = 100s$). (g) Imagem do sensor de visão (Ep. 80, $t = 100s$); (h) Mapa Atencional normalizado (Ep. 80, $t = 102s$); (i) Mapa de Saliências (Ep. 80, $t = 102s$); (j) Característica vencedora do ciclo atencional (Ep. 80, $t = 102s$).

A Figura 28 apresenta os valores médios e os desvios padrão do número de ações executado a cada 10 episódios ao longo deste experimento. A Figura 29 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo deste experimento. O maior número de ações executadas (305) é obtido no episódio 100. O número médio de ações executadas a cada 10 episódios se mantém acima de 60 desde o início deste experimento.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

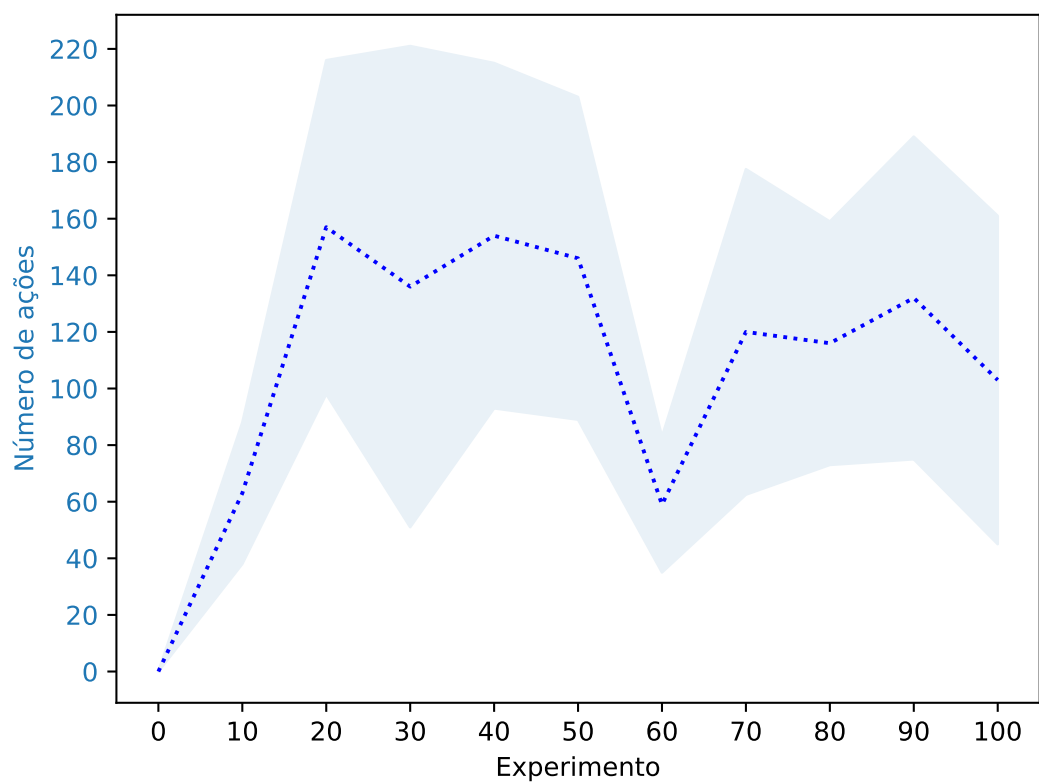


Figura 28 – (Berto 1º Subestágio - Exp. B) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando lentamente e de cor primária.

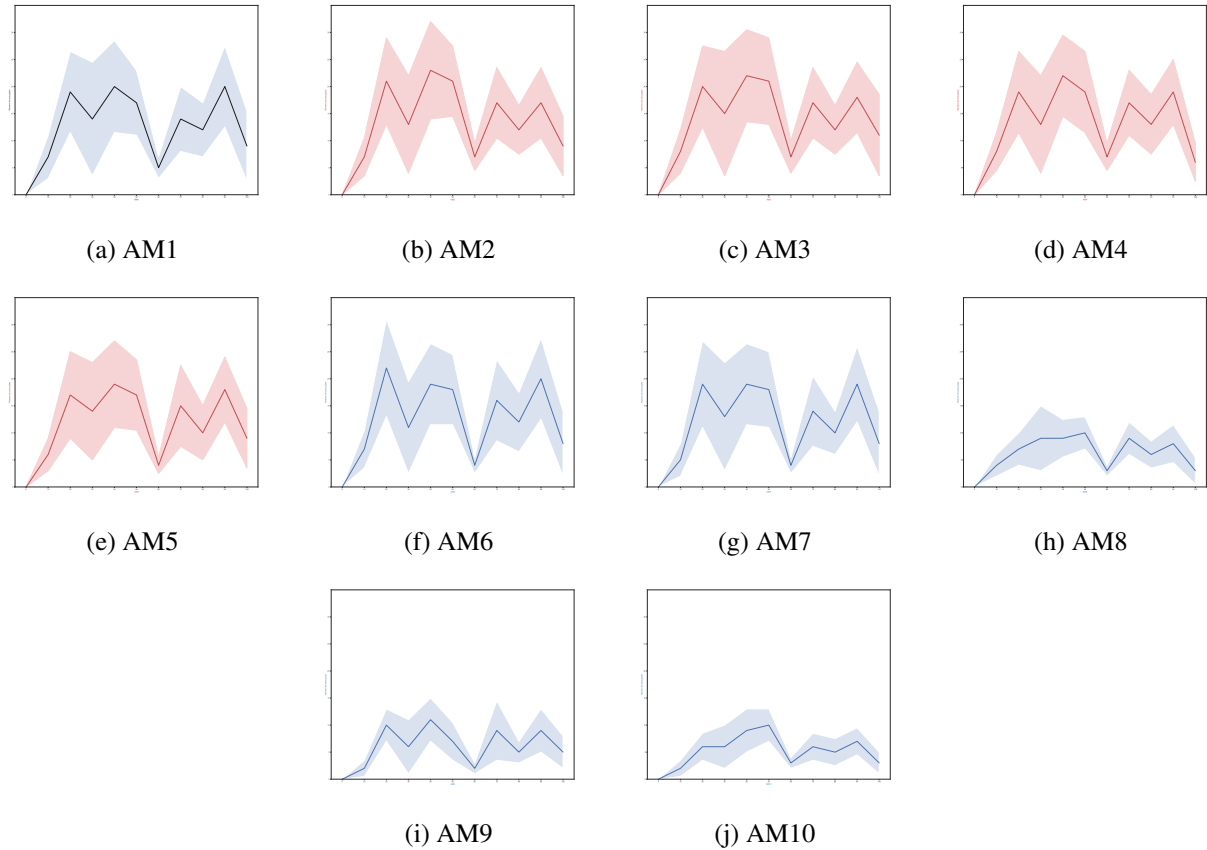


Figura 29 – (Berto - 1º Subestágio - Exp. B) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. Objeto se movimentando lentamente e de cor primária: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea.

5.2 (EXP-02) 2º Subestágio: Reações Circulares Primárias

Neste experimento é utilizada a humanoide Marta com um algoritmo cognitivo-atencional que possui somente os elementos em azul e vermelho demonstrados na Figura 19. A humanoide Marta continua com apenas elementos de percepção *bottom-up*. Com a implementação de um modelo de motivação, a agente passa a explorar suas possíveis ações que não possuem esquemas definidos na Memória Procedimental M_p . As reações reflexas desenvolvidas no primeiro estágio podem gerar reações circulares primárias mediante a estabilização do aprendizado de determinados conjuntos de ações. Para simular o aumento na acuidade visual deste subestágio, adotou-se uma resolução para o sensor de visão de 128 x 128 pixels. A resposta da agente foi investigada para a presença de objetos estáticos com diferentes cores e posições e um robô Pioneer 3DX, que possui um algoritmo de Braitenberg implementado e se move com uma velocidade média de 0.05 m/s, se comportando como um corpo móvel no ambiente. Foram executados 200 episódios durante o Aprendizado Procedimental. Ao fim de cada episódio, a recompensa adquirida e o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer 3DX é redefinida aleatoriamente. O experimento demonstrou o curso atencional da seleção para percepção para uma agente executando somente o curso *bottom-up* da atenção com elementos motivacionais.

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações; (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

5.2.1 Validação do Sistema Atencional

A cena do simulador *CoppeliaSim* no instante $t=8s$ no 1º episódio é demonstrada na Figura 30 (a). A imagem obtida pelo sensor de visão no instante $t=8s$ no 1º episódio é demonstrada na Figura 30 (b). Os resultados do ciclo atencional para os estímulos obtidos nos instantes seguintes são demonstrados nas Figuras 30 (c-n).

Os resultados do ciclo atencional deste subestágio são demonstrados na Figura 30. Através da ferramenta de comunicação externa *RemoteAPI CoppeliaSim*, o sensor de visão da Humanoide Marta coleta informações sensoriais (intensidades de cores nos três canais RGB e distâncias relativas ao sensor de visão) do ambiente, formando objetos de memória que correspondem à estas observações, representados como $Color_{MO}$ e $Depth_{MO}$. Estes objetos são armazenadas em quantidade limitada na memória sensorial pelos codelets *Color Buffer* e *Depth Buffer*, gerando os objetos de memória $ColorBuffer_{MO}$ e $DepthBuffer_{MO}$. Os níveis em RGB dos canais de cor vermelho, verde e azul para os pixels disponibilizados pelo sensor de visão no instante $t=8s$ do 1º episódio são demonstrados nas Figuras 30 (c), (d) e (e), respectivamente. As distâncias relativas (0: muito próximo, 10: muito distante) ao sensor de visão para o instante $t=8s$ do 1º episódio são demonstradas na Figura 30 (f). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=8s$ do 1º episódio, em escala de cinza.

Por meio dos objetos de memória $Color_{MO}$ e $Depth_{MO}$, são construídos os mapas de características *bottom-up* de intensidades de cores e de distância pelos codelets *Red bottom-up FeatureMap*, *Green bottom-up FeatureMap*, *Blue bottom-up FeatureMap* e *Distance bottom-up FeatureMap*, gerando mapas de características representados como objetos de memória: $RedFM_{BUMO}$, $GreenFM_{BUMO}$, $BlueFM_{BUMO}$ e $DistanceFM_{BUMO}$. Os mapas de características *bottom-up* para os canais de cor vermelho, verde e azul no instante $t=9s$ do 1º episódio são demonstrados nas Figuras 30 (g), (h) e (i). O mapa de características *bottom-up* para as distâncias relativas ao sensor de visão no instante $t=9s$ do 1º episódio são demonstradas na Figura 30 (j). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=8s$ do 1º episódio, em escala de cinza.

Os mapas de características são combinados em um mapa de características combinado pelo codelet *CombinedFeatureMap*, conforme proposto pelo Modelo de Colombini (2016) (COLOMBINI, 2014), gerando o objeto de memória $CombFM_{MO}$. O Mapa Atencional é inicializado com valores unitários, simulando um ambiente sem vencedores, e é

realimentado a cada iteração pelo codelet de Tomada de Decisão *Decision Making*. O mapa de saliências será então obtido pelo codelet *SaliencyMap* através do produto entre o Mapa de Características Combinado ($CombFM_{MO}$) e o Mapa Atencional ($AttMap_{MO}$), também seguindo o Modelo de Colombini. O mapa de saliências $SalMap_{MO}$ influencia o codelet de Tomada de Decisão, definindo a característica vencedora de cada ciclo, que influenciará no decorrer do ciclo atencional, com base nos ciclos inibitórios e excitatórios da operação *bottom-up*. O mapa de características combinado, o Mapa Atencional normalizado, o Mapa de Saliências e a Característica vencedora no instante $t=10s$ do 1º episódio são demonstrados nas Figuras 30 (k), (l), (m) e (n). Conforme demonstram as imagens, os mapas realçam as regiões sensoriais de interesse à agente. Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=8s$ do 1º episódio, em escala de cinza.

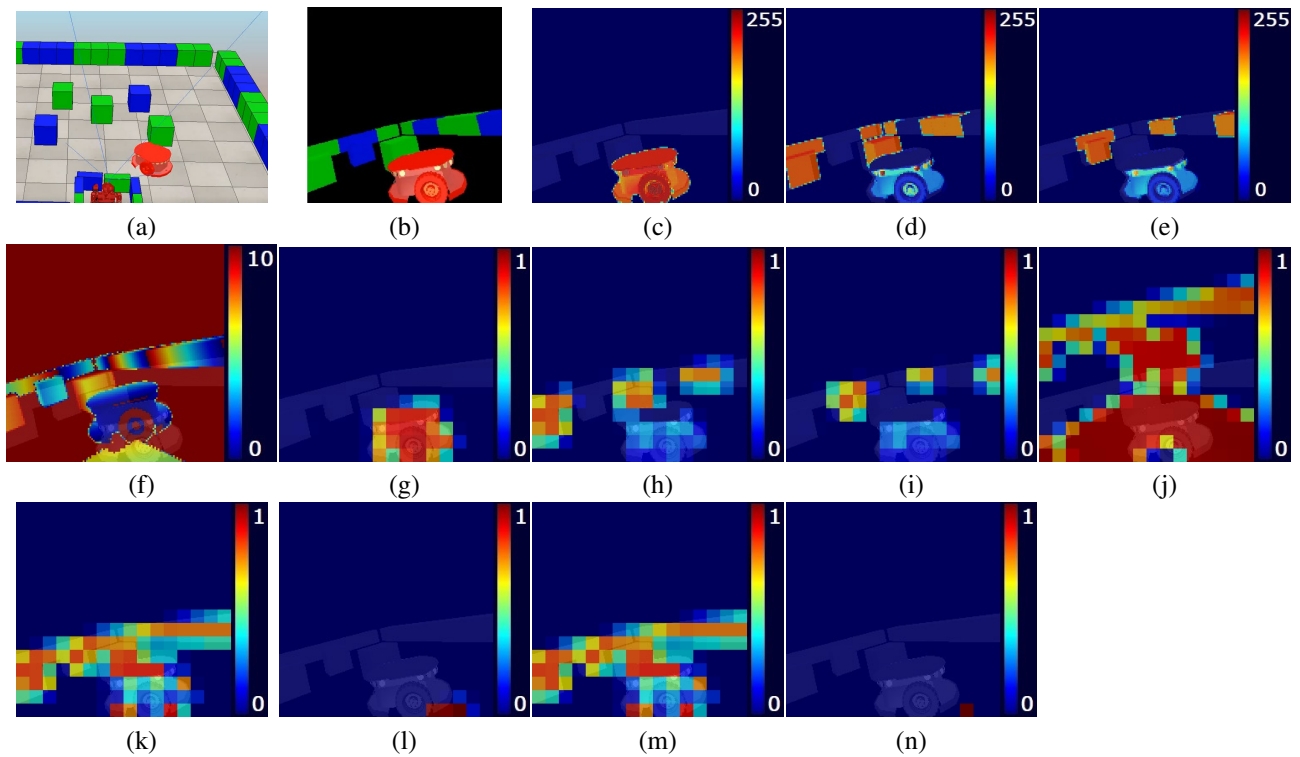


Figura 30 – (2º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental - (a) Cena no simulador *CoppeliaSim* ($t = 8s$); (b) Imagem do sensor de visão ($t = 8s$); (c) Valores do canal vermelho para o sensor de visão ($t = 8s$); (d) Valores do canal verde para o sensor de visão ($t = 9s$); (e) Valores do canal azul para o sensor de visão ($t = 8s$); (f) Distância relativa ao sensor de visão ($t = 8s$); (g) *Bottom-up Feature Map* para o canal vermelho ($t = 9s$); (h) *Bottom-up Feature Map* para o canal verde ($t = 9s$); (i) *Bottom-up Feature Map* para o canal azul ($t = 9s$); (j) *Bottom-up Feature Map* para distância ($t = 9s$); (k) Mapa de características combinado ($t = 10s$); (l) Mapa Atencional normalizado ($t = 10s$); (m) Mapa de Saliências ($t = 10s$); (n) Característica vencedora do ciclo atencional ($t = 10s$).

5.2.2 Validação do Sistema de Aprendizado Procedimental

A Figura 31 apresenta os valores médios e os desvios padrão de recompensas obtidos pela agente e o número médio de ações a cada 10 episódios ao longo do Aprendizado Procedimental. A Figura 32 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo do Aprendizado Procedimental.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

A utilização da QTable do subestágio anterior permite à agente uma melhor performance. Nos primeiros episódios, a Motivação da agente incentiva a exploração de todo novo esquema não existente na Memória Procedimental M_p para o refinamento das ações reflexas desenvolvidas no subestágio anterior e a formação de reações circulares primárias. Além de promover a execução de todas as ações estabelecidas através da Função de Motivação, a Tomada de Decisão da agente promove um aumento no posicionamento do atuador *NeckYaw* para a direita, em acompanhamento do deslocamento do robô Pioneer P3DX.

O afastamento do Pioneer P3DX no episódio 10 provoca o aumento na execução de a_{M4} , demonstrando o interesse da agente com objetos próximos e seu próprio corpo, em decorrência dos reflexos reforçados durante a exploração destas regiões no subestágio anterior. A reaproximação do Pioneer P3DX no instante $t=10s$ do episódio 20 promove o direcionamento do sensor de visão da agente para o corpo externo e a identificação do Pioneer P3DX como a característica vencedora do ciclo atencional, conforme demonstram as Figuras 33 (a-e). A partir do episódio 90, o número médio de ações e recompensas se mantém acima de 100. Os valores máximos de ações executadas (429) e recompensa adquirida (560) são obtidos no episódio 197.

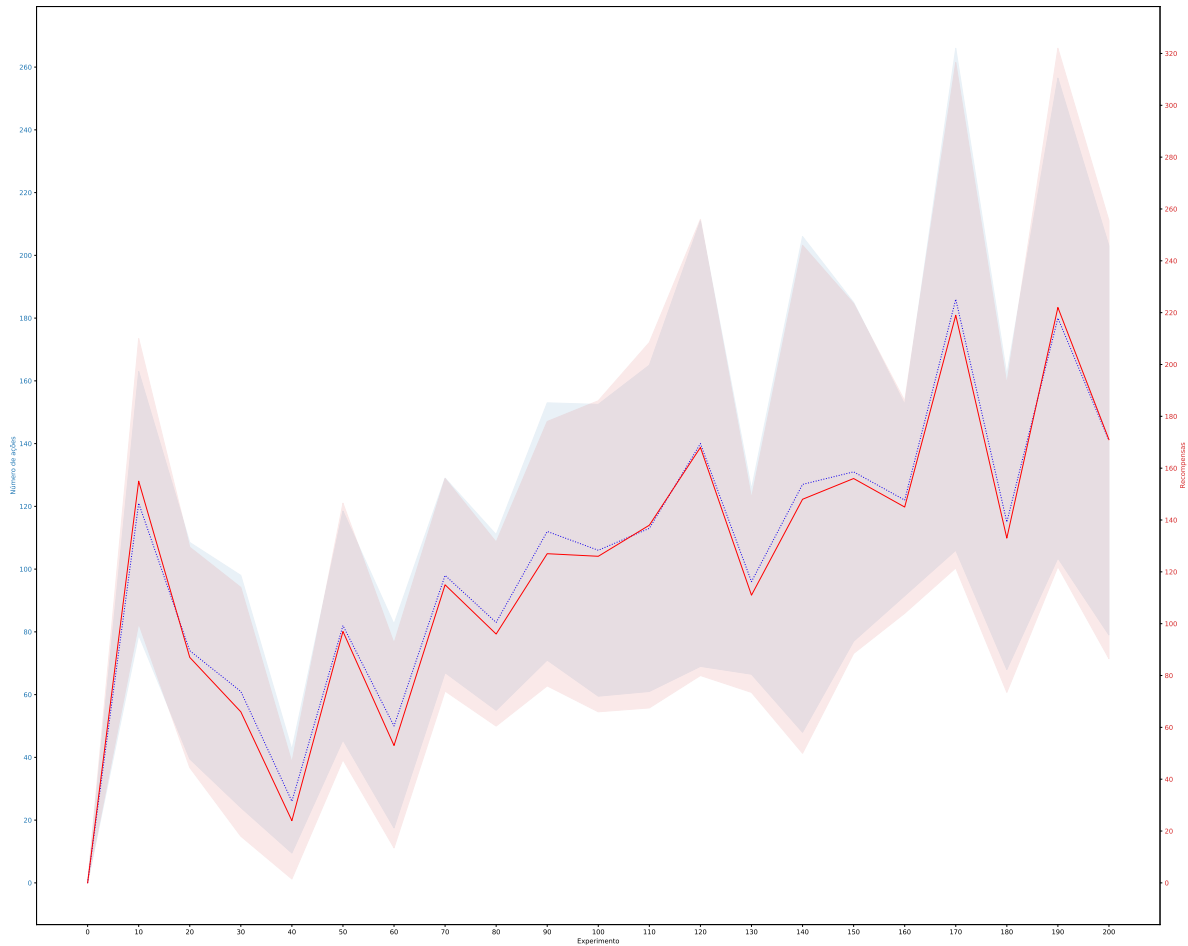


Figura 31 – (2º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número total de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.

Neste subestágio, a agente minimizou a atuação de seus reflexos relacionadas aos estímulos obtidos com a exploração de seu corpo e aos blocos próximos à sua localização, limitando o tempo em que se mantém focada nestas regiões

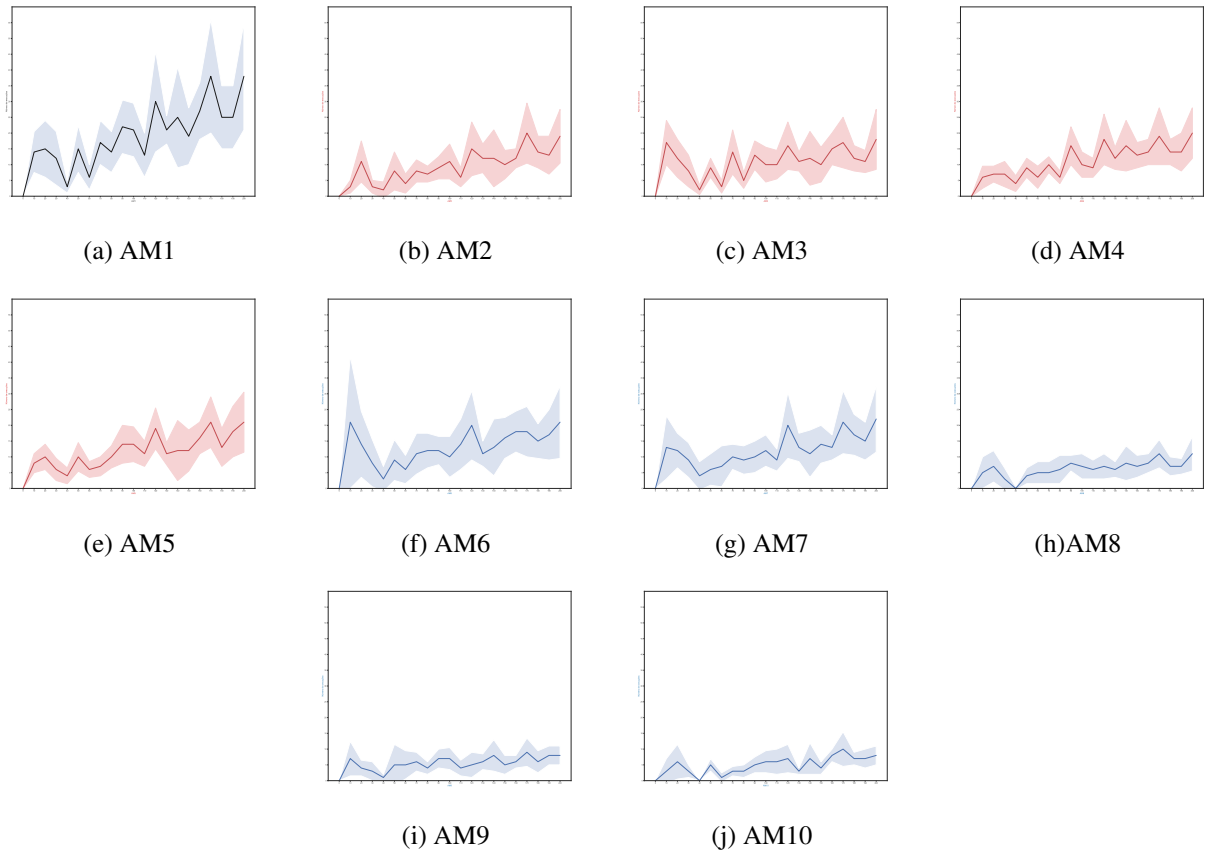


Figura 32 – (2º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios do Aprendizado Procedimental: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea.

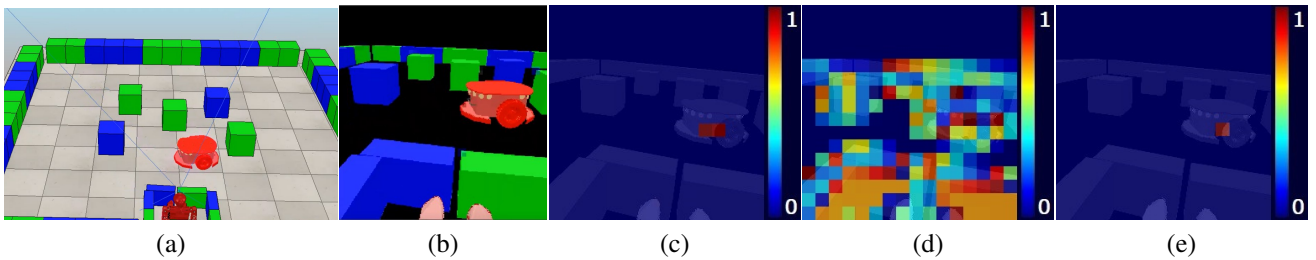


Figura 33 – (2º Subestágio) Dados sensoriais obtidos no episódio 20 do Aprendizado Procedimental - (a) Cena no simulador *CoppeliaSim* ($t = 30s$); (b) Imagem do sensor de visão ($t = 30s$); (c) Mapa Atencional normalizado ($t = 32s$); (d) Mapa de Saliências ($t = 32s$); (e) Característica vencedora do ciclo atencional ($t = 32s$).

quando o robô Pioneer P3DX se afasta. A agente também promoveu a formação de reações circulares primárias com a execução das ações de movimentação a_{M2-6} e de atualizar a posição da fóvea a_{M6-10} em conjunto com a ação a_{M1} manter o foco, promovendo a formação de reações circulares primárias entre com procedimentos que incluem estas

ações. Ao final dos episódios do 2º subestágio, a tabela QTable construída pelo *Q-learning* representa a receita de execução das tarefas aprendidas e armazenadas. Os procedimentos poderão ser recordados no próximo subestágio, bem como melhorados por novas ações da agente.

5.2.3 Validação do Aprendizado com os Experimentos de Berto - Objeto se movimentando e de cor primária

De forma a verificar o aprendizado da agente durante este subestágio, foi utilizada a tabela QTable resultante do fim do último episódio do Aprendizado Procedimental para realizar o seguinte experimento, adaptado dos experimentos cognitivos propostos por Berto (2020) para a avaliação de agentes cognitivos para o 2º subestágio sensório-motor da Teoria de Jean Piaget (PIAGET, 1952) na tarefa de rastrear objetos. Os experimentos propostos por Berto (2020) são apresentados no Anexo A. Foram realizadas 100 execuções de, no máximo, 500 ações cada. Ao fim de cada episódio, o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer P3DX é redefinida aleatoriamente.

A robô Marta foi posicionada com o robô Pioneer P3DX fora de seu campo de visão, conforme ilustra a cena da Figura 34 (a). O robô Pioneer P3DX possui um algoritmo de Braitenberg e se movimenta com uma velocidade constante de 0.1m/s em todas as execuções deste experimento. O sistema atencional da agente identifica o robô Pioneer P3DX como ponto mais saliente e o sistema de Tomada de Decisão promove o rastreamento de sua movimentação, conforme demonstram as Figuras 34 (b-j). Os mapas sobrepõem a imagem obtida pelo sensor de visão em escala de cinza.

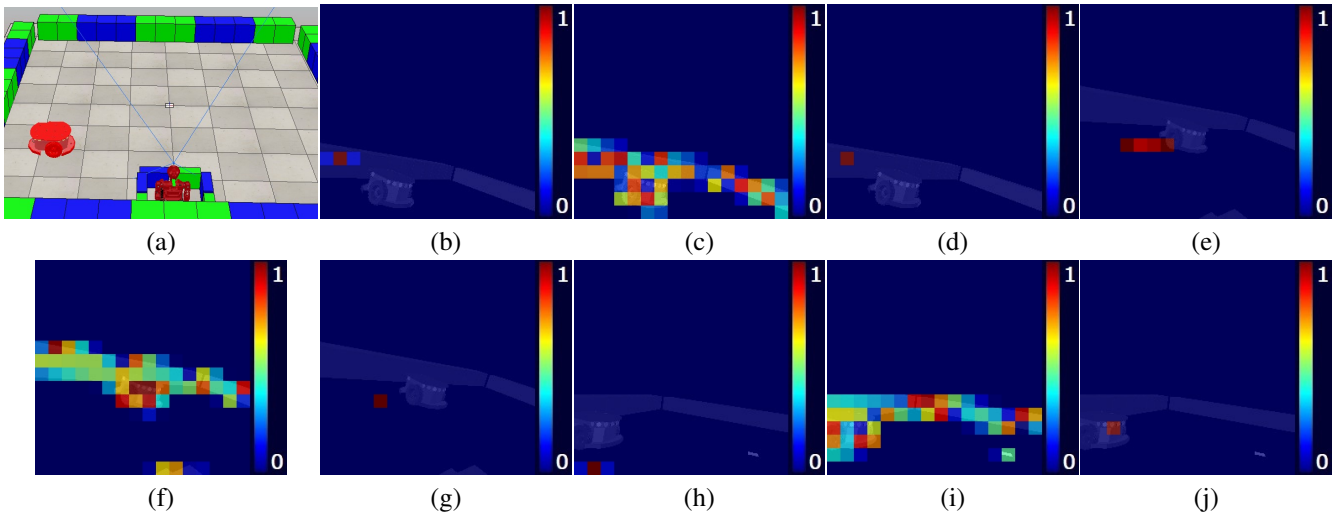


Figura 34 – (Berto - 2º Subestágio) Dados sensoriais obtidos nas execuções deste experimento. Objeto se movimentando e de cor primária - (a) Cena do *CoppeliaSim* (Ep. 1, $t = 1s$); (b) Mapa Atencional normalizado (Ep. 1, $t = 32s$); (c) Mapa de Saliências (Ep. 1, $t = 32s$); (d) Característica vencedora do ciclo atencional (Ep. 1, $t = 32s$); (e) Mapa Atencional normalizado (Ep. 60, $t = 21s$); (f) Mapa de Saliências (Ep. 60, $t = 21s$); (g) Característica vencedora do ciclo atencional (Ep. 60, $t = 21s$); (h) Mapa Atencional normalizado (Ep. 100, $t = 13s$); (i) Mapa de Saliências (Ep. 100, $t = 13s$); (j) Característica vencedora do ciclo atencional (Ep. 100, $t = 13s$).

A Figura 35 apresenta os valores médios e os desvios padrão do número de ações executado a cada 10 episódios ao longo deste experimento. A Figura 36 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo deste experimento. O maior número de ações executadas (207) é obtido no episódio 95. O número médio de ações a cada 10 episódios se mantém acima de 100 desde o início deste experimento.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

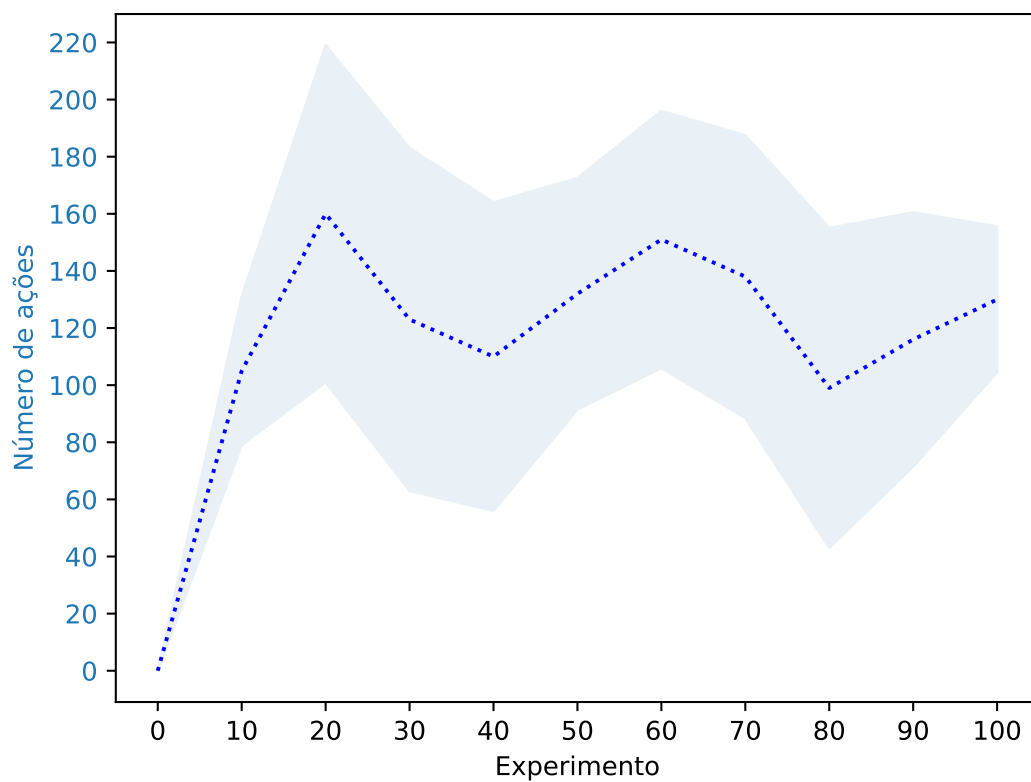


Figura 35 – (Berto - 2º Subestágio) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando e de cor primária. Objeto se movimentando e de cor primária.

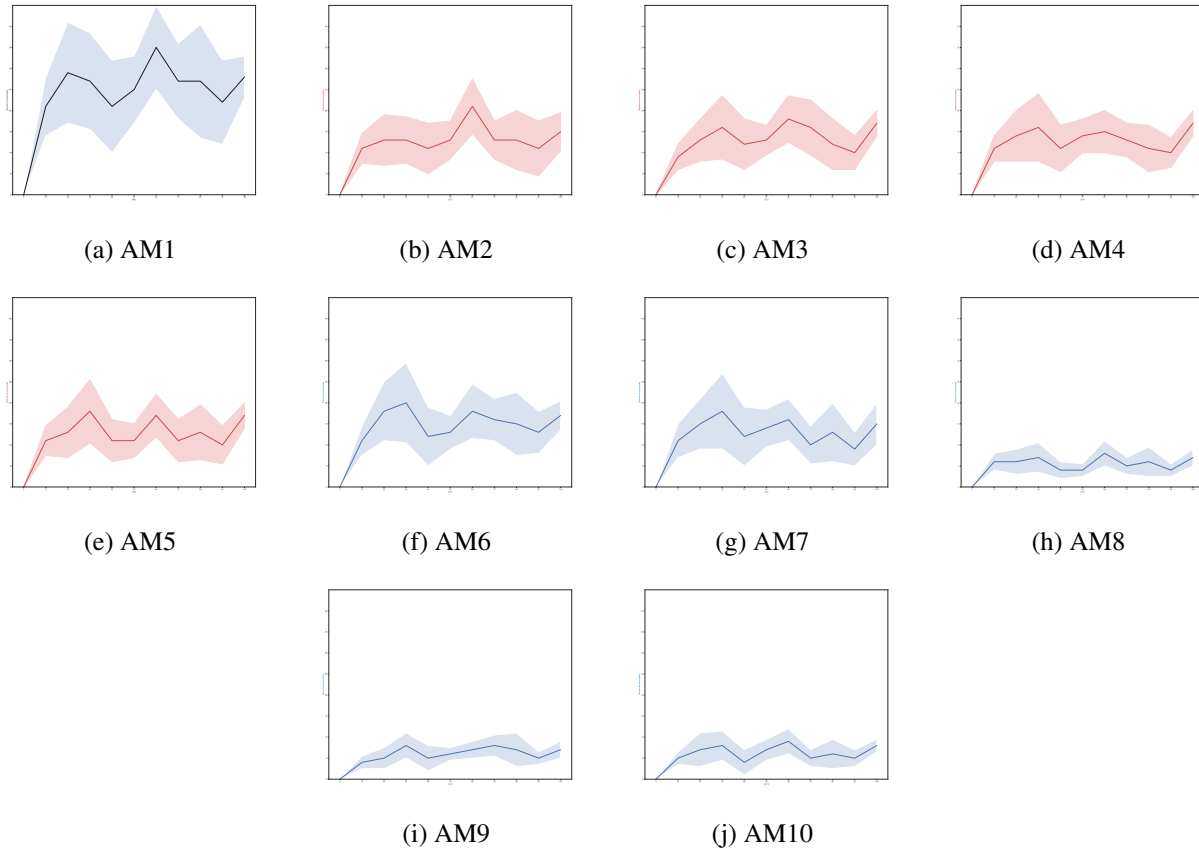


Figura 36 – (Berto - 2º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios. Objeto se movimentando e de cor primária: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea.

5.3 (EXP-03) 3º Subestágio: Reações Circulares Secundárias

Neste subestágio é utilizada a humanoide Marta com um algoritmo cognitivo-atencional que possui todos os elementos demonstrados na Figura 19. A humanoide Marta possui elementos de percepção *bottom-up* e *top-down*. Com a implementação de modelos de intencionalidade e motivação, a agente passa a explorar as reações circulares primárias desenvolvidas nos estágios anteriores e a desenvolver reações circulares secundárias. Para simular o aumento na acuidade visual deste subestágio, adotou-se uma resolução para o sensor de visão de 256 x 256 pixels. A resposta da agente será investigada para a presença de objetos estáticos com diferentes cores e posições e um robô Pioneer 3DX, que possui um algoritmo de Braitenberg implementado e se move com uma velocidade média de 0.05 m/s, se comportando como um corpo móvel no ambiente. O Aprendizado Procedimental foi executado durante 200 episódios. Ao fim de cada episódio, a recompensa adquirida e o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer 3DX é redefinida aleatoriamente.

Cada tentativa de execução (episódio) neste experimento é executada até a agente: (i) atingir o número máximo de passos/ações; (ii) cair ou ultrapassar os limites de seus atuadores motores; ou (iii) se não obter saliências em seu ciclo atencional por 5 iterações seguidas.

5.3.1 Validação do Sistema Atencional

Os resultados do ciclo atencional deste subestágio são demonstrados na Figura 37. Através da ferramenta de comunicação externa *RemoteAPI CoppeliaSim*, o sensor de visão da Humanoide Marta coleta informações sensoriais (intensidades de cores nos três canais RGB e distâncias relativas ao sensor de visão) do ambiente, formando objetos de memória que correspondem à estas observações, representados como $Color_{MO}$ e $Depth_{MO}$. Estes objetos são armazenados em quantidade limitada na memória sensorial pelos codelets *Color Buffer* e *Depth Buffer*, gerando os objetos de memória $ColorBuffer_{MO}$ e $DepthBuffer_{MO}$. Neste subestágio, a resolução estabelecida pelo atuador virtual para o sensor de visão é de 256 x 256 pixels. Os níveis em RGB dos canais de cor vermelho, verde e azul para os pixels disponibilizados pelo sensor de visão no instante $t=1s$ do 1º episódio são demonstrados nas Figuras 37 (c), (d) e (e), respectivamente. As distâncias relativas (0m: muito próximo, 10m: muito distante) ao sensor de visão para o instante $t=1s$ do 1º episódio são demonstradas na Figura 37 (f). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza.

Por meio dos objetos de memória $Color_{MO}$ e $Depth_{MO}$, são construídos os mapas de características *bottom-up* de intensidades de cores e de distância pelos codelets *Red bottom-up FeatureMap*, *Green bottom-up FeatureMap*, *Blue bottom-up FeatureMap* e *Distance bottom-up FeatureMap*, gerando mapas de características representados como objetos de memória: $RedFM_{BUMO}$, $GreenFM_{BUMO}$, $BlueFM_{BUMO}$ e $DistanceFM_{BUMO}$. Os mapas de características *bottom-up* para os canais de cor vermelho, verde e azul no instante $t=2s$ do 1º episódio são demonstrados nas Figuras 37 (g), (h) e (i). O mapa de características *bottom-up* para as distâncias relativas ao sensor de visão no instante $t=2s$ do 1º episódio são demonstradas na Figura 37 (j). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza.

A intencionalidade da agente neste subestágio promove a formação dos mapas de características *top-down* de cor desejada, distância desejada e região desejada. Estes mapas são representados pelos codelets *Color top-down FeatureMap*, *Depth top-down FeatureMap* e *Region top-down FeatureMap*, gerando mapas de características representados como objetos de memória: $ColorFM_{TDMO}$, $DepthFM_{TDMO}$ e $RegionFM_{TDMO}$. Os mapas de características *top-down* para: cor desejada: Vermelho (R: 255, G: 0, B: 0); distância desejada (muito próximo: 0m) e região desejada (centro: 64) no instante $t=2s$ do 1º episódio são demonstrados nas Figuras 37 (k), (l) e (m). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza.

Os mapas de características *bottom-up* e *top-down* são combinados em um mapa de características combinado pelo codelet *CombinedFeatureMap*, conforme proposto pelo Modelo de Colombini (2016) (COLOMBINI, 2014), gerando o objeto de memória *CombFM_{MO}*. O Mapa Atencional é inicializado com valores unitários, simulando um ambiente sem vencedores, e é realimentado a cada iteração pelo codelet de Tomada de Decisão *Decision Making*. O mapa de saliências será então obtido pelo codelet *SaliencyMap* através do produto entre o Mapa de Características Combinado (*CombFM_{MO}*) e o Mapa Atencional (*AttMap_{MO}*), também seguindo o Modelo de Colombini. O mapa de saliências *SalMap_{MO}* influencia o codelet de Tomada de Decisão, definindo a característica vencedora de cada ciclo, que influenciará no decorrer do ciclo atencional. Neste subestágio, os ciclos inibitórios e excitatórios da operação *bottom-up* só atuam sobre as características vencedoras decorrentes de estímulos *bottom-up*. O mapa de características combinado, o Mapa Atencional normalizado, o Mapa de Saliências e a Característica vencedora no instante $t=3s$ do 1º episódio são demonstrados nas Figuras 30 (n), (o), (p) e (q). Conforme demonstram as imagens, os mapas realçam as regiões sensoriais de interesse à agente. Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=1s$ do 1º episódio, em escala de cinza.

Inicialmente, o Mapa Atencional é inicializado com valores unitários. Neste cenário, a característica vencedora do ciclo atencional foi a região central 64, decorrente dos estímulos *top-down* de região desejada. No entanto, na primeira alteração do Mapa Atencional no instante $t=5s$ pelo codelet de Tomada de Decisão promove a alteração da característica vencedora do ciclo atencional para a região 45, decorrente dos estímulos *top-down* de cor-desejada (vermelho), conforme ilustram as Figuras 38 (a), (b) e (c).

5.3.2 Validação do Sistema de Aprendizado Procedimental

A Figura 39 apresenta os valores médios e os desvios padrão de recompensas obtidos pela agente e o número médio de ações a cada 10 episódios ao longo do Aprendizado Procedimental. A Figura 40 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo do Aprendizado Procedimental.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

A utilização da QTable do subestágio anterior permite à agente uma melhor performance. Nos primeiros episódios, observa-se a atuação das reações circulares primárias relacionadas à ação a_{M1} (manter o foco) em conjunto com as ações a_{M2-5} (movimentação dos atuadores *NeckYaw* e *HeadPitch*) para acompanhar o robô Pioneer 3DX. Com o afastamento do robô Pioneer 3DX, a Função de Motivação incentiva o posicionamento do foco atencional de acordo com o Mapa de Características *top-down* de Distância desejada (muito distante: 10m) ou de Cor-desejada (Vermelho - R: 255, G:0, B:0), evitando o posicionamento para as regiões dos estímulos advindos de objetos próximos e promovendo a formação de reações circulares secundárias com as ações a_{A1} , com a Cor-desejada (Vermelho - R: 255, G:0, B:0), e a_{A2} , com a Distância desejada (muito distante: 10m).

A reaproximação do robô Pioneer 3DX incentiva a formação de reações circulares secundárias das ações a_{A1} , com a Cor-desejada (Vermelho - R: 255, G:0, B:0), a_{A2} , com a Distância desejada (muito próximo: 0m) e a_{A3} , com a Região desejada (centro: 64). A identificação do robô Pioneer 3DX através do posicionamento da fóvea a partir do Mapa de Características *top-down* de distância desejada (muito distante: 10m) é ilustrada nas Figuras (a-c). Os mapas sobrepõem a imagem obtida pelo sensor de visão no instante $t=32s$ do episódio 180, em escala de cinza.

As ações a_{M2-5} (movimentação dos atuadores *NeckYaw* e *HeadPitch*) são executadas em concordância com as ações de a_{M6-10} de posicionamento da fóvea. A agente utiliza das ações a_{M11-14} (movimentação dos atuadores *NeckYaw* e *HeadPitch* para perto ou longe do foco atencional) em conjunto com as ações atencionais a_{A1-3} para percorrer as regiões próximas à característica vencedora do ciclo atencional de forma a encontrar estados em que a recompensa recebida é maximizada. A partir do episódio 80, o número médio de ações e recompensas se mantém

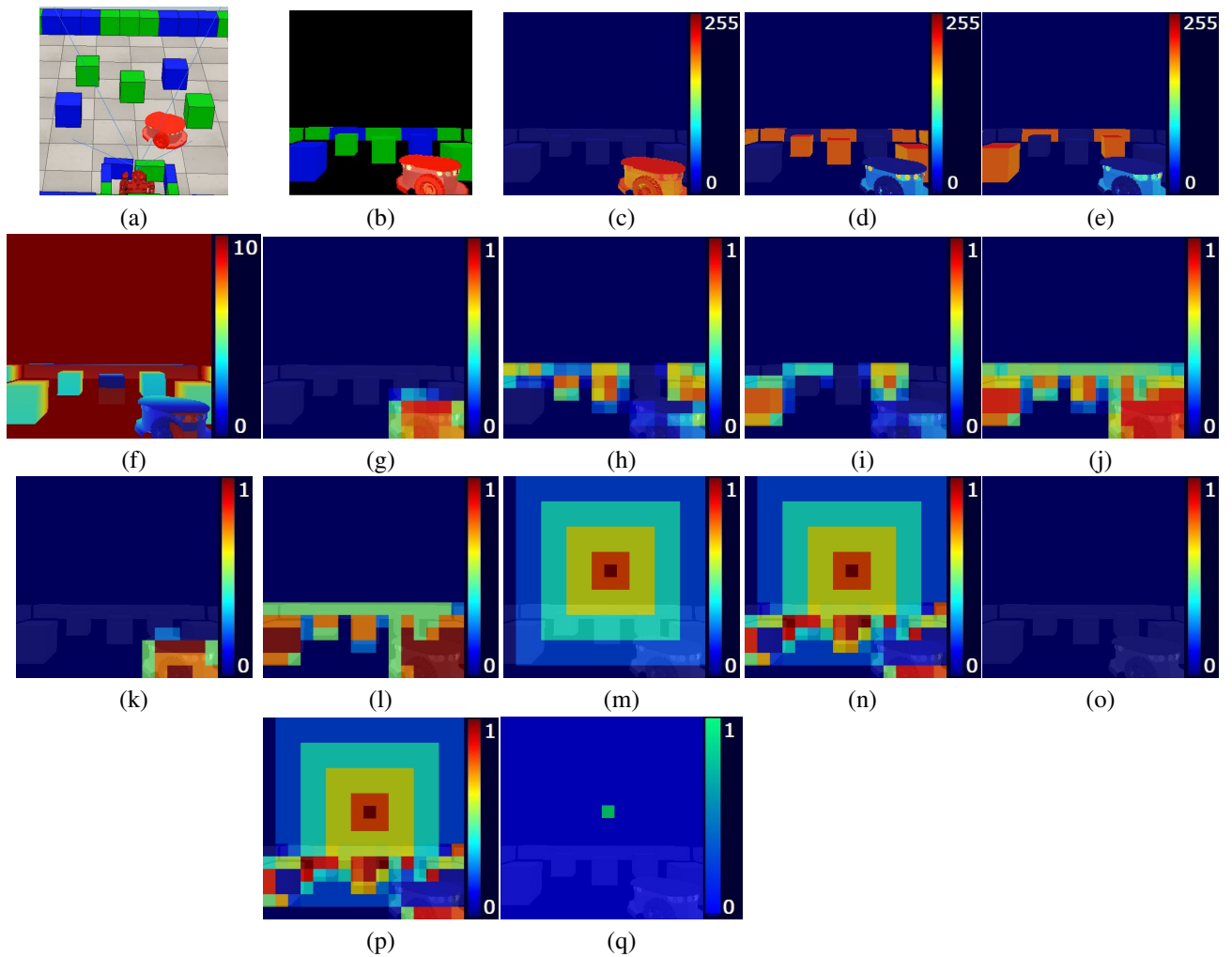


Figura 37 – (3º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental - (a) Cena do *CoppeliaSim* (Ep. 1, $t = 1s$); (b) Imagem do sensor de visão ($t = 1s$); (c) Valores do canal vermelho para o sensor de visão ($t = 1s$); (d) Valores do canal verde para o sensor de visão ($t = 1s$); (e) Valores do canal azul para o sensor de visão ($t = 1s$); (f) Distância relativa ao sensor de visão ($t = 1s$); (g) *Bottom-up Feature Map* para o canal vermelho ($t = 2s$); (h) *Bottom-up Feature Map* para o canal verde ($t = 2s$); (i) *Bottom-up Feature Map* para o canal azul ($t = 2s$); (j) *Bottom-up Feature Map* para distância ($t = 2s$); (k) *Top-down Feature Map* para cor desejada (Vermelho. R:255, G:0, B:0) ($t = 2s$); (l) *Top-down Feature Map* para distância desejada (muito próximo: 0m) ($t = 2s$); (m) *Top-down Feature Map* para região desejada (centro: 64) ($t = 2s$); (n) Mapa de características combinado ($t = 3s$); (o) Mapa Atencional ($t = 3s$); (p) Mapa de Saliências ($t = 3s$); (q) Característica vencedora do ciclo atencional: *top-down* ($t = 3s$).

acima de 100. Os valores máximos de ações executadas (429) e recompensa adquirida (560) são obtidos no episódio 197.

Neste subestágio, a agente refinou a atuação de suas reações circulares primárias e promoveu a formação de reações circulares secundárias através do monitoramento dos estímulos *top-down*. Ao final dos episódio do 3º subestágio, a tabela QTable construída pelo *Q-learning* representa a receita de execução das tarefas aprendidas e armazenadas.

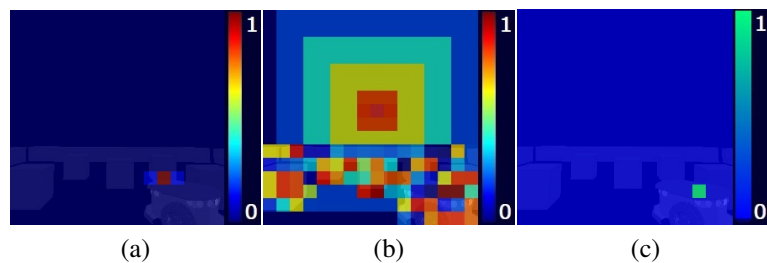


Figura 38 – (3º Subestágio) Dados sensoriais obtidos no 1º episódio do Aprendizado Procedimental - (a) Mapa Atencional ($t = 5s$); (b) Mapa de Saliências ($t = 5s$); (c) Característica vencedora do ciclo atencional: *top-down* ($t = 5s$).

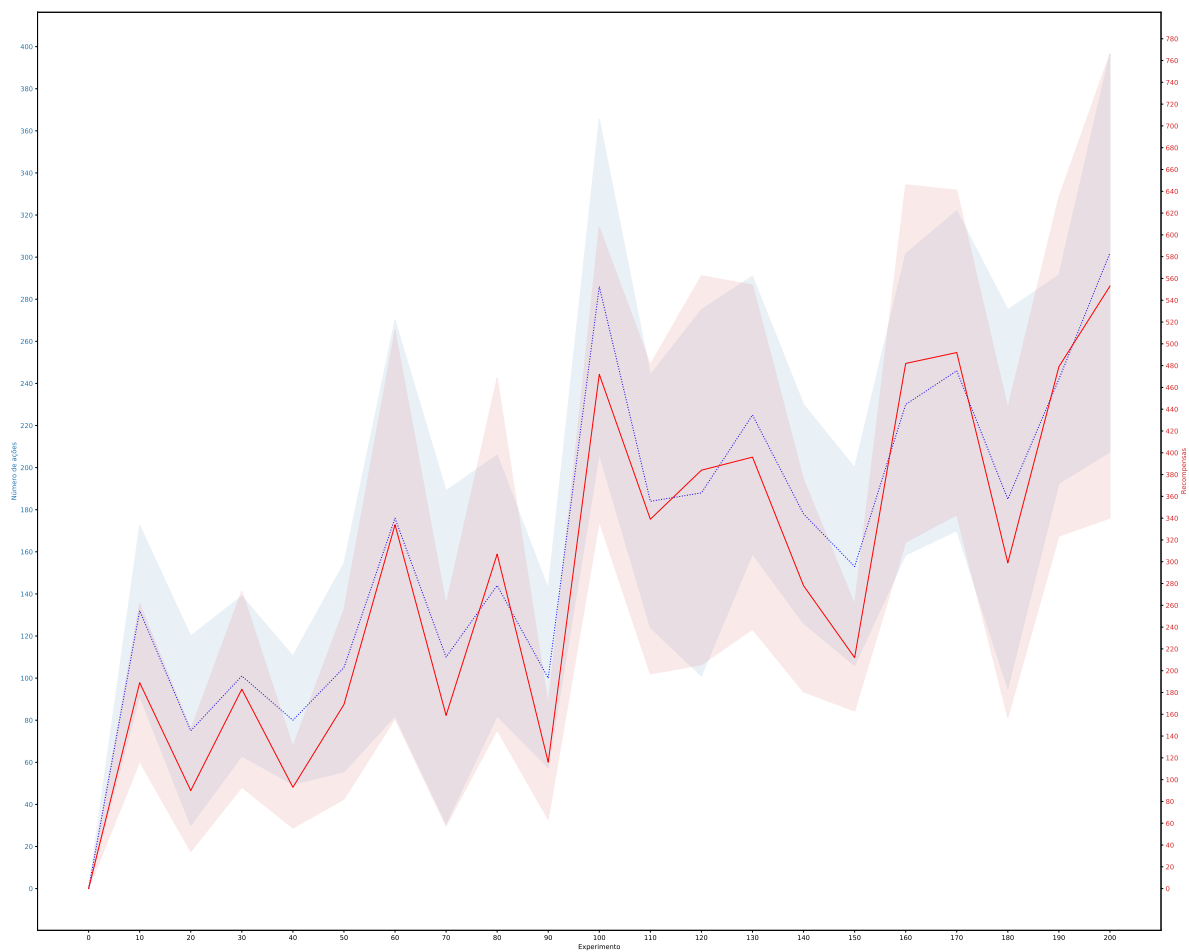


Figura 39 – (3º Subestágio) Médias e desvios padrão das recompensas adquiridas (linha contínua - vermelha) e número total de ações executadas (linha pontilhada - azul) a cada 10 episódios do Aprendizado Procedimental.

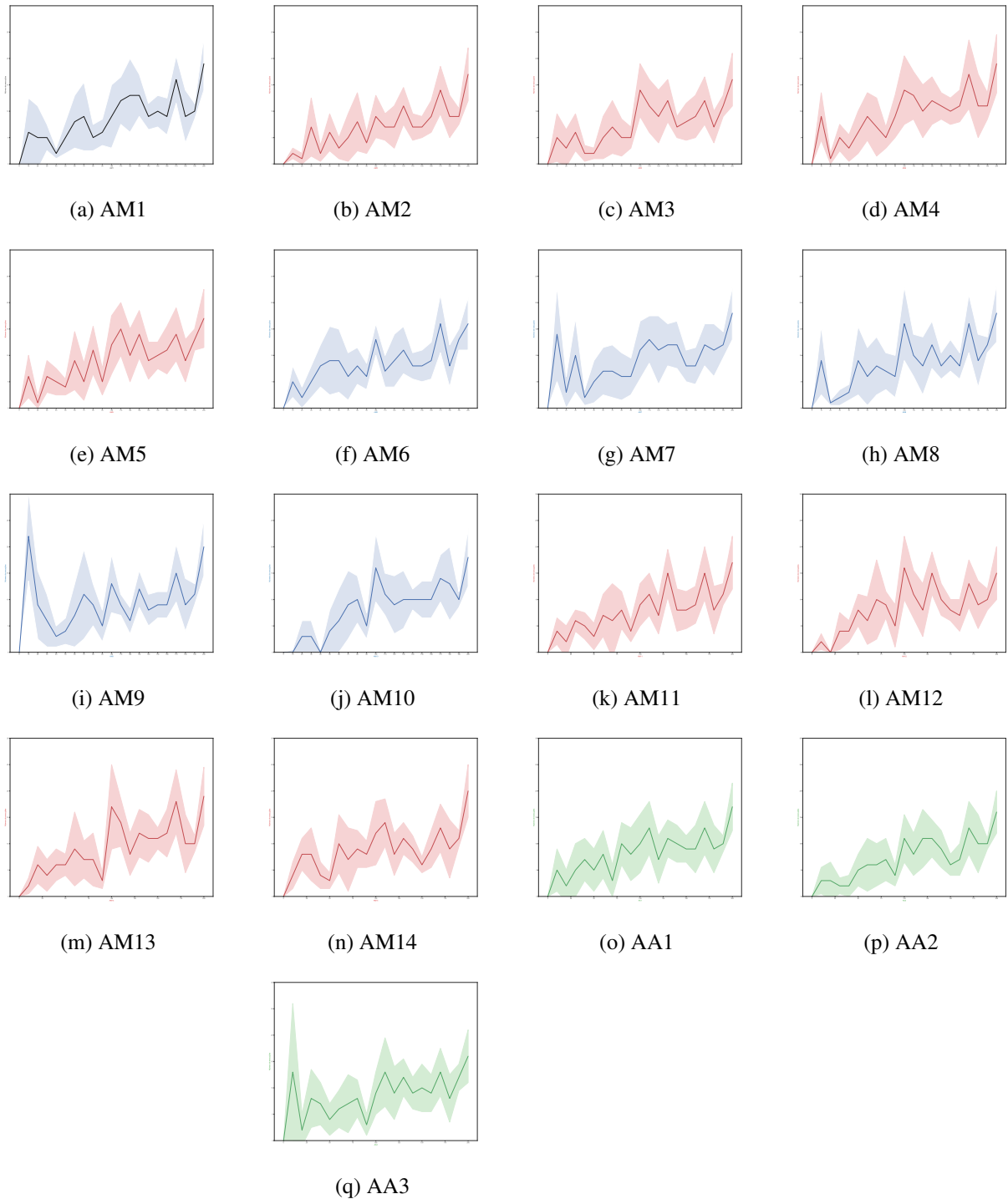


Figura 40 – (3º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios no Aprendizado Procedimental: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4); (k) a_{M11} (*NeckYaw* p/ foco atencional); (l) a_{M12} (*NeckYaw* p/ longe do foco atencional); (m) a_{M13} (*HeadPitch* p/ foco atencional); (n) a_{M14} (*HeadPitch* p/ longe do foco atencional); (o) a_{A1} (mover foco atencional de acordo com $\mathcal{F}_5$); (p) a_{A2} (mover foco atencional de acordo com $\mathcal{F}_6$); (q) a_{A3} (mover foco atencional de acordo com $\mathcal{F}_7$). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea; **Média em verde, desvio verde:** ações atencionais.

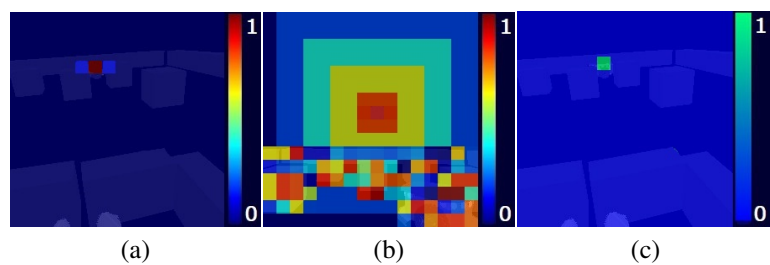


Figura 41 – (3º Subestágio) Dados sensoriais obtidos no episódio 180 do Aprendizado Procedimental - (a) Mapa Atencional ($t = 34s$); (b) Mapa de Saliências ($t = 34s$); (c) Característica vencedora do ciclo atencional: *top-down* ($t = 34s$).

5.3.3 Validação do Aprendizado com os Experimentos de Berto - Objeto se movimentando e de cor primária

De forma a verificar o aprendizado da agente durante este subestágio, foi utilizada a tabela QTable resultante do fim do último episódio do Aprendizado Procedimental para realizar o seguinte experimento, adaptado dos experimentos cognitivos propostos por Berto (2020) para a avaliação de agentes cognitivos para o 3º subestágio sensório-motor da Teoria de Jean Piaget (PIAGET, 1952) na tarefa de rastrear objetos. Os experimentos propostos por Berto (2020) são apresentados no Anexo A. Foram realizadas 100 execuções de, no máximo, 500 ações cada. Ao fim de cada episódio, o número de ações executadas é zerado, os atuadores da humanoide Marta retornam para a posição inicial e a posição do robô Pioneer 3DX é redefinida aleatoriamente.

A robô Marta foi posicionada com o robô Pioneer 3DX fora de seu campo de visão, conforme ilustram as cenas das Figuras 42 (a), (b), (c) e (d). O robô Pioneer 3DX possui um algoritmo de Braitenberg e se movimenta com uma velocidade constante de 0.1m/s em todas as execuções deste experimento. O sistema atencional da agente identifica o robô Pioneer 3DX como ponto mais saliente e o sistema de Tomada de Decisão promove o rastreamento de sua movimentação, conforme demonstram as Figuras 42 (c-e). Os mapas sobrepõem a imagem obtida pelo sensor de visão em escala de cinza.

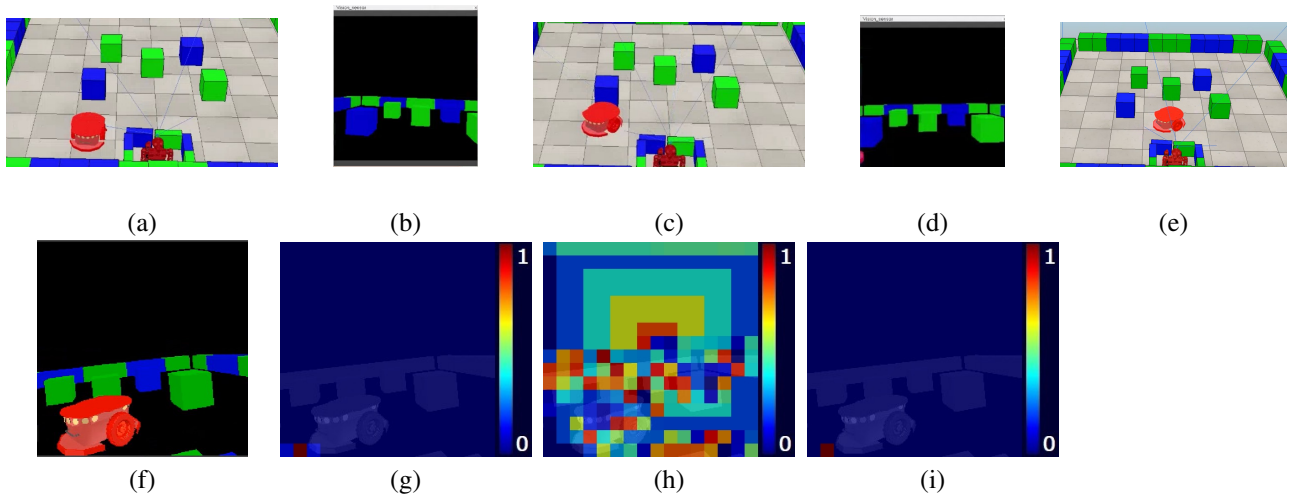


Figura 42 – (Berto - 3º Subestágio) Dados sensoriais obtidos na primeira execução deste experimento. Objeto se movimentando e de cor primária. - (a) Cena do *CoppeliaSim* (Ep. 1, $t = 1s$); (b) Imagem obtida pelo sensor de visão (Ep. 1, $t = 1s$); (c) Cena do *CoppeliaSim* (Ep. 1, $t = 4s$); (d) Imagem obtida pelo sensor de visão (Ep. 1, $t = 4s$); (e) Cena do *CoppeliaSim* (Ep. 1, $t = 47s$); (f) Imagem obtida pelo sensor de visão (Ep. 1, $t = 47s$); (g) Mapa Atencional normalizado (Ep. 1, $t = 49s$); (h) Mapa de Saliências (Ep. 1, $t = 49s$); (i) Característica vencedora do ciclo atencional (Ep. 1, $t = 49s$).

A Figura 43 apresenta os valores médios e os desvios padrão do número de ações executado a cada 10 episódios ao longo deste experimento. A Figura 44 apresenta os valores médios e os desvios padrão do número de cada possível ação executada a cada 10 episódios ao longo deste experimento. O maior número possível de ações executadas (500) é obtido em 11 episódios. O número médio de ações a cada 10 episódios se mantém acima de 100 desde o início deste experimento.

Os valores absolutos utilizados na confecção dos gráficos de número de ações executadas para cada possível ação são apresentados no Anexo B.

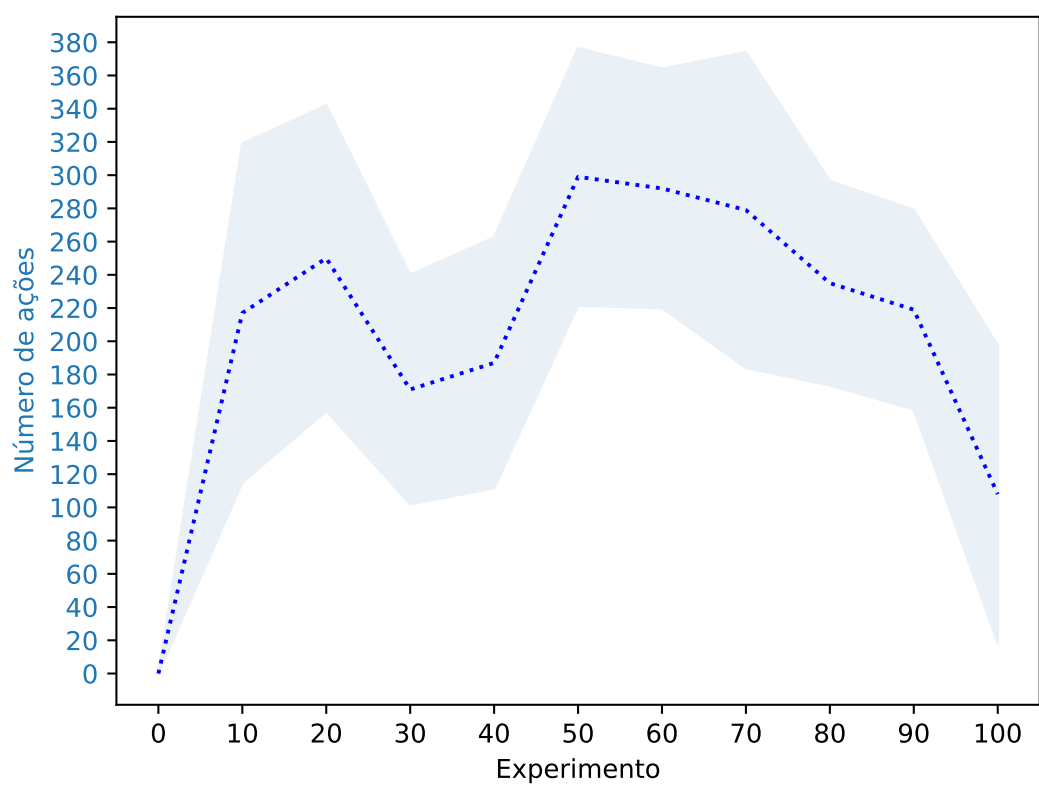


Figura 43 – (Berto - 3º Subestágio) Médias e desvios padrão do número de ações executadas a cada 10 episódios. Objeto se movimentando e de cor primária.

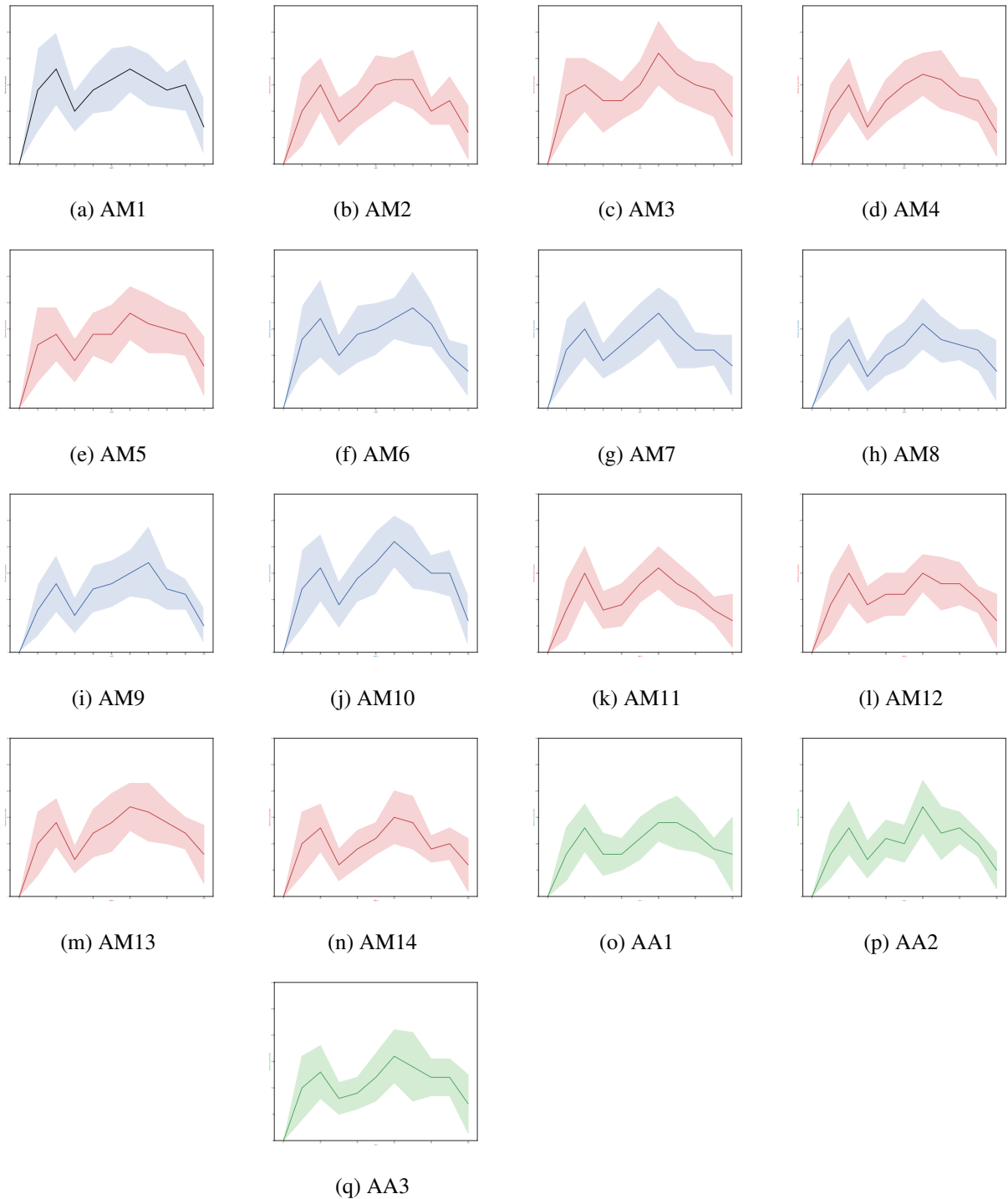


Figura 44 – (Berto - 3º Subestágio) Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios: (a) a_{M1} (manter o foco); (b) a_{M2} (*NeckYaw* p/ esquerda); (c) a_{M3} (*NeckYaw* p/ direita); (d) a_{M4} (*HeadPitch* p/ baixo); (e) a_{M5} (*HeadPitch* p/ cima); (f) a_{M6} (Fóvea p/ posição 0); (g) a_{M7} (Fóvea p/ posição 1); (h) a_{M8} (Fóvea p/ posição 2); (i) a_{M9} (Fóvea p/ posição 3); (j) a_{M10} (Fóvea p/ posição 4); (k) a_{M11} (*NeckYaw* p/ foco atencional); (l) a_{M12} (*NeckYaw* p/ longe do foco atencional); (m) a_{M13} (*HeadPitch* p/ foco atencional); (n) a_{M14} (*HeadPitch* p/ longe do foco atencional); (o) a_{A1} (mover foco atencional de acordo com $\mathcal{F}_5$); (p) a_{A2} (mover foco atencional de acordo com $\mathcal{F}_6$); (q) a_{A3} (mover foco atencional de acordo com $\mathcal{F}_7$). **Média em preto, desvio azul:** manter o foco; **Média em vermelho, desvio vermelho:** ações de movimentação dos atuadores motores; **Média em azul, desvio azul:** ações de posicionamento do atuador virtual da fóvea; **Média em verde, desvio verde:** ações atencionais.

5.4 Discussões

Os valores obtidos pelo Sistema Atencional da agente durante o Aprendizado Procedimental permitiram a tomada de decisão de acordo com o foco atencional e demonstraram características de cada subestágio piagetiano. Os valores médios de recompensas adquiridas e ações executadas nos experimentos voltados ao Aprendizado Procedimental demonstraram o crescimento incremental no aprendizado da agente com a reutilização dos esquemas aprendidos em subestágios anteriores. Os valores médios de ações executadas para cada possível ação nos experimentos voltados ao Aprendizado Procedimental demonstraram as preferências por determinadas ações pelo *codelet* de Tomada de Decisão da agente com estados em particular. A Figura 45 apresenta o padrão de execução de um episódio de teste em cada subestágio na qual a agente se encontra.

Para o 1º subestágio é apresentado o episódio 84 do experimento de Berto B. Neste experimento, o robô Pioneer se movimenta como um corpo móvel no ambiente. A agente executa ações de movimentação em sequência e executa poucas vezes as ações a_{M1} manter o foco e a_{M6-10} movimentação da fóvea. O distanciamento do robô Pioneer na ação 20 promove a execução das ações a_{M3-5} movimentação do atuador *HeadPitch* para cima e para baixo e do atuador *NeckYaw* para a direita, acompanhando o robô Pioneer. No entanto, quando o Pioneer passa para o fundo da cena, o foco atencional é direcionado para o corpo da agente e os atuadores motores são direcionados para esta região. Com o retorno do robô Pioneer, a agente retornaram o sensor de visão ao campo externo. A execução de diferentes ações de movimentação em sequência promoveu a ocorrência de ruídos nos mapas de saliências, dificultando o processo de tomada de decisão da agente a partir dos estados adquiridos. Apesar de executar poucas ações de movimentação da fóvea, a agente demonstra coerência entre a execução destas ações e das ações de movimentação dos atuadores motores.

Para o 2º subestágio é apresentado o episódio 83 do experimento de Berto. Neste experimento, o robô Pioneer se movimenta como um corpo móvel no ambiente. A agente demonstrou padrões nas execuções de ações a_{M2-5} de movimentação dos atuadores motores, a_{M6-10} movimentação da fóvea e a_{M1} manter o foco. A agente aumentou consideravelmente a execução da ação a_{M1} manter o foco e das ações a_{M6-10} movimentação da fóvea. Este comportamento permite uma identificação mais precisa dos estados. Com o distanciamento do Pioneer para o fundo da cena, o foco atencional é direcionado para o corpo da agente. No entanto, neste caso, as ações de movimentação dos atuadores motores são pouco executados, não direcionando o sensor de visão para esta região. Para este cenário, foram mais executadas as ações de a_{M1} manter o foco e a_{M6-7} movimentação da fóvea. Especula-se que a baixa execução das ações a_{M8-10} movimentação da fóvea é ocasionada pela não identificação do robô Pioneer quando este se encontra nas regiões mais distantes.

Para o 3º subestágio é apresentado o episódio 82 do experimento de Berto. Neste experimento, o robô Pioneer se movimenta como um corpo móvel no ambiente e cubos estáticos de cores diferentes podem promover a oclusão do Pioneer. A agente promoveu a formação de reações circulares secundárias. A agente executa as ações, em sequência: (i) a_{M1} manter o foco; (ii) a_{A1-3} ações atencionais; (iii) a_{M6-10} movimentação da fóvea e, por fim, (iv) a_{M2-5} de movimentação. Neste lote de experimentos, com a movimentação do Pioneer para o fundo da cena, o foco atencional é direcionado com as ações atencionais a_{A1} cor desejada: vermelho e a_{A3} distância desejada: 10. A agente demonstrou a execução de padrões coordenados na execução das ações.

5.4.1 1º Subestágio

O Sistema Atencional do 1º Subestágio, durante o Aprendizado Procedimental, permitiu à agente estabelecer seu foco atencional no robô Pioneer P3DX, enquanto o mesmo se movimentava nas regiões mais próximas à humanoide. Com o afastamento do robô Pioneer P3DX, a agente orientou seu foco atencional para objetos próximos e para seu próprio

corpo. Os estímulos obtidos durante a exploração do corpo da agente reforçaram os reflexos utilizados. A ausência de um sistema de motivação fez com que a agente executasse em maior quantidade as ações reforçadas (a_{M1-7}), mesmo quando os estímulos que promoveram este reforço não estavam mais presentes, e em menor quantidade as ações que não participaram destas interações (a_{M8-10}). A agente também demonstrou a habilidade de integrar reflexos, de forma a aumentar sua recompensa adquirida, entre: (i-iv) as ações de movimentação a_{M2-5} e a ação de manter foco a_{M1} ; (v-vi) as ações de atualizar a posição da fóvea a_{M6-7} e a ação de manter foco a_{M1} .

Nas execuções do experimento A para este subestágio, adaptado dos experimentos de Berto (2020), a agente direcionou inicialmente seu foco atencional para o robô Pioneer 3DX, que manteve-se parado durante este experimento, conforme sugerido por Berto (2020). No entanto, a atuação dos ciclos excitatórios e inibitórios promovidos pelo Sistema Atencional direcionaram novamente o foco atencional para as regiões mais próximas da humanoide. A atuação dos reflexos reforçados durante a exploração do corpo da agente no Aprendizado Procedimental acarretaram na alternância dos atuadores da agente entre as regiões mais próximas da humanoide e o robô Pioneer 3DX.

Nas execuções do experimento B para este subestágio, adaptado dos experimentos de Berto (2020), a agente direcionou seu foco atencional para o robô Pioneer 3DX, enquanto este manteve-se nas regiões mais próximas da agente (região frontal) durante este experimento. No entanto, com o afastamento do robô Pioneer 3DX para as áreas laterais, a atuação dos reflexos reforçados durante o Aprendizado Procedimental acarretaram novamente no direcionamento do foco atencional para as regiões mais próximas da humanoide.

5.4.2 2º Subestágio

O Sistema Atencional do 2º Subestágio, durante o Aprendizado Procedimental, permitiu à agente estabelecer seu foco atencional no robô Pioneer 3DX, enquanto o mesmo se movimentava nas regiões mais próximas à humanoide, como no 1º Subestágio. Com o afastamento do robô Pioneer 3DX, a agente novamente orientou seu foco atencional para objetos próximos e para seu próprio corpo. No entanto, com a atuação do sistema de motivação, a agente foi motivada a explorar todas as ações possíveis a cada novo esquema não encontrado na Memória Procedimental. Este comportamento minimizou a atuação das ações reforçadas (a_{M1-7}). A utilização dos reflexos em conjunto (i-iv) com as ações de movimentação a_{M2-5} e a ação de manter foco a_{M1} ; (v-vi) as ações de atualizar a posição da fóvea a_{M6-7} e a ação de manter foco a_{M1}), desenvolvidos no 1º subestágio, permitiram à agente adquirir maiores recompensas e promover a formação de reações circulares primárias. Este comportamento faz com que a ação a_{M1} seja a mais executada. Com a baixa exploração das regiões mais distantes do ambiente, as ações a_{M8-10} continuaram sendo menos executadas.

Nas execuções do experimento adaptado dos experimentos de Berto (2020) para este subestágio, a agente manteve seu foco atencional para o robô Pioneer 3DX enquanto o mesmo se movimentava próximo à localização da agente e estava dentro de seu campo de visão. A agente direcionou seu foco atencional para as regiões mais próximas de seu corpo com o afastamento do robô Pioneer 3DX para fora do seu campo de visão, conforme proposto por Berto (2020). A utilização das reações circulares primárias desenvolvidas durante este subestágio promoveram um desempenho com números maiores de ações executadas em comparação ao subestágio anterior. O aumento da acuidade visual e refinamento da movimentação dos atuadores neste subestágio permitem à agente um maior controle em seus atuadores com relação ao subestágio anterior.

5.4.3 3º Subestágio

O Sistema Atencional do 3º Subestágio, durante o Aprendizado Procedimental, permitiu à agente estabelecer seu foco atencional no robô Pioneer 3DX. Neste subestágio, mesmo com o afastamento do robô Pioneer 3DX, o sistema de motivação incentivou a exploração de todas as ações possíveis a cada novo esquema não encontrado na Memória

Procedimental. Com a utilização das ações atencionais a_{A1-3} , a agente foi capaz de acompanhar a movimentação do robô Pioneer P3DX mesmo quando ele se encontrava nas regiões mais distantes à humanoide. Neste cenário, com o distanciamento do robô Pioneer P3DX, a agente promoveu a formação de reações circulares secundárias com as ações a_{A1} , com a Cor-desejada (Vermelho - R: 255, G:0, B:0), e a_{A2} , com a Distância desejada (muito distante: 10m). Já com a reaproximação do robô Pioneer P3DX, a agente promoveu a formação de reações circulares secundárias das ações a_{A1} , com a Cor-desejada (Vermelho - R: 255, G:0, B:0), a_{A2} , com a Distância desejada (muito próximo: 0m) e a_{A3} , com a Região desejada (centro: 64).

Nas execuções do experimento adaptado dos experimentos de [Berto \(2020\)](#) para este subestágio, a agente manteve seu foco atencional para o robô Pioneer P3DX mesmo com seu distanciamento, conforme proposto por [Berto \(2020\)](#). Este comportamento permitiu a agente executar diferentes ações de forma mais uniforme, com o intuito de receber mais recompensas com as diferentes ações executadas. O aumento da acuidade visual neste subestágio e refinamento da movimentação dos atuadores permitem à agente um maior controle em seus atuadores com relação aos subestágios anteriores. A utilização das reações circulares secundárias desenvolvidas durante este subestágio promoveram um desempenho com números maiores de ações executadas em comparação aos subestágios anteriores.

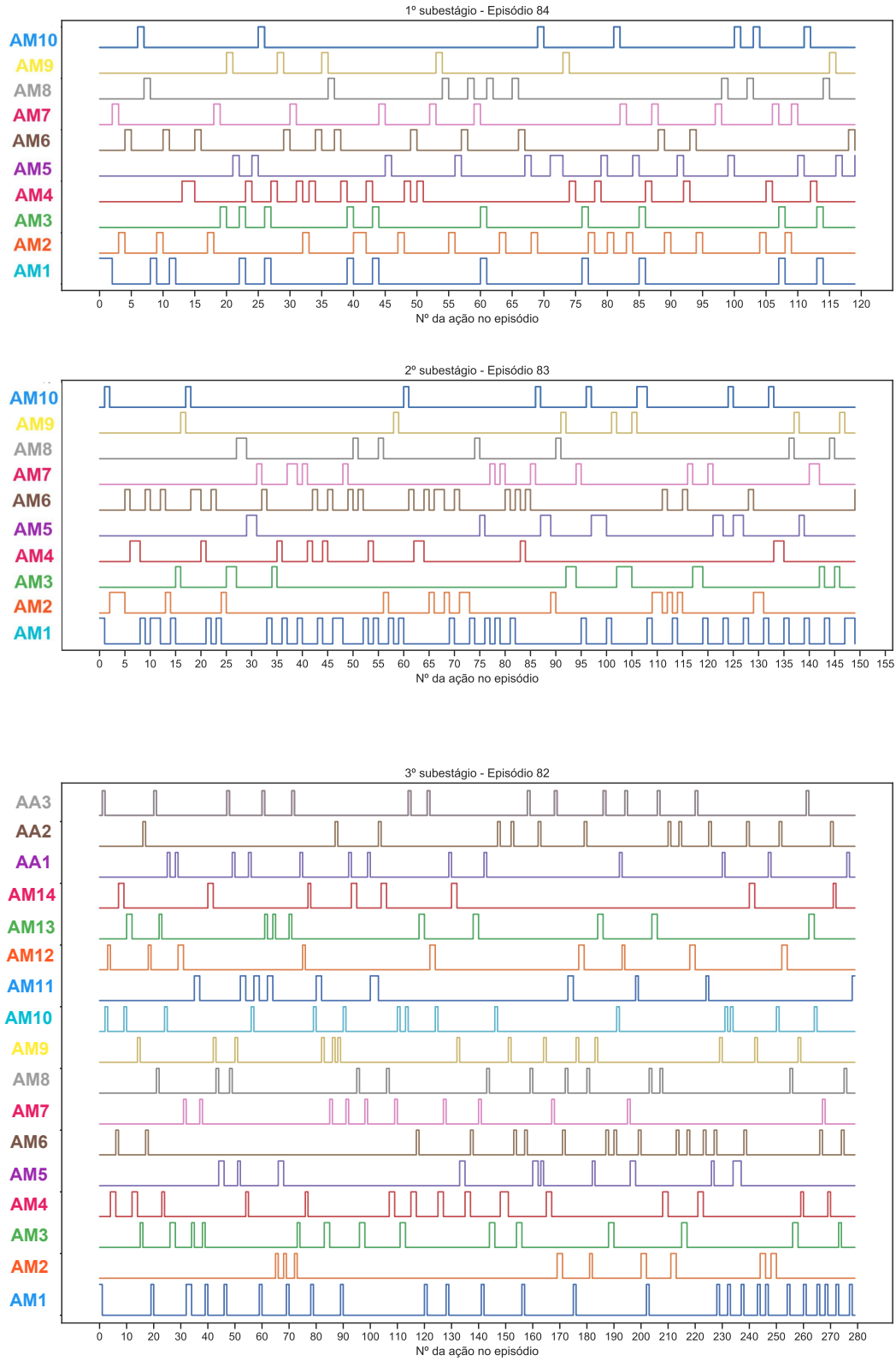


Figura 45 — Mudanças de ações executadas em um episódio teste em cada subestágio. 1º Subs.: Ep. 84 - Exp. B; 2º Subs.: Ep. 83; 3º Subs.: Ep. 82; Ações Motoras: (1º e 2º subestágios) a_{M1} manter o foco; a_{M2} NeckYaw p/ esquerda; a_{M3} NeckYaw p/ direita; a_{M4} HeadPitch p/ baixo; a_{M5} HeadPitch p/ cima; a_{M6} Fóvea p/ posição 0; a_{M7} Fóvea p/ posição 1; a_{M8} Fóvea p/ posição 2; a_{M9} Fóvea p/ posição 3; a_{M10} Fóvea p/ posição 4; (3º subestágio) a_{M11} NeckYaw p/ foco atencional; a_{M12} NeckYaw p/ longe do foco atencional; a_{M13} HeadPitch p/ foco atencional; a_{M14} HeadPitch p/ longe do foco atencional. Ações atencionais: (3º subestágio) a_{A1} mover foco atencional de acordo com $\mathcal{F}_5$; a_{A2} mover foco atencional de acordo com $\mathcal{F}_6$; a_{A3} mover foco atencional de acordo com $\mathcal{F}_7$.

6 CONCLUSÃO

"Nós só podemos ver um pouco do futuro, mas o suficiente para perceber que há muito a fazer."

(Alan Turing)

6.1 Considerações Finais

Os experimentos realizados demonstraram a viabilidade da utilização de uma arquitetura cognitiva-atencional modelada com ferramentas de robótica cognitiva, proposta para agentes conscientes computando sobre o paradigma atencional. Além das características apresentadas, diversas outras podem ser incorporadas ou aprimoradas. Uma importante característica desta abordagem é que este modelo de sistema atencional provê aos níveis cognitivos de mais alta hierarquia espaços de estados densos, que no caso do experimento apresentado, foi utilizado como base para o sistema de aprendizado, ao invés do sistema de percepção tradicional. Prontamente, nestes experimentos, a utilização desse espaço atencional promoveu uma redução de 262.144 (65.536 pixels de 4 *byte-arrays* - RGB e *Depth*) para 16 estados.

Nas implementações foi utilizado um sensor de visão e foram definidas as dimensões das características relacionados às tarefas de rastreamento de objetos propostos por Berto (2020) para o desenvolvimento cognitivo na área de robótica. Os resultados apresentados demonstram que a arquitetura de integração proposta é capaz de modelar os elementos essenciais dos processos atencionais e utilizá-los com um processo de aprendizado incremental em um domínio da robótica autônoma. A seleção do componente de percepção foi utilizada como base do processo de tomada de decisão, aplicada com sucesso nas tarefas consideradas. Particularmente, a exploração natural no ambiente quando a característica vencedora advém de um estímulo exógeno, promovida pelo efeito IOR, e a competição ou o alinhamento dos efeitos atencionais *bottom-up* e *top-down* constroem a observação do ambiente, sendo útil na aplicação de redução do espaço de estados para componentes de Tomada de decisão e Aprendizado.

A implementação de um sistema de Aprendizado Procedimental capaz de adquirir (assimilar) e refinar (acomodar) conhecimento procedimental (estados, ações e recompensas) possui inspiração no aprendizado de esquemas, conforme proposto por Jean Piaget. Prontamente, arquitetura implementada atua como uma máquina de processamento de esquemas.

A reutilização dos procedimentos aprendidos em subestágios anteriores permitiu à agente refinar seus reflexos inatos e construir procedimentos mais complexos. No 2º subestágio, o sistema de motivação foi capaz de minimizar os reflexos reforçados durante o 1º subestágio durante a exploração do corpo da agente. Já no 3º subestágio, a combinação do sistema de motivação com a utilização de elementos capazes de interpretar estímulos *top-down* e a atuação de ações atencionais no 3º subestágio permitiram à agente desenvolver reações circulares secundárias relacionadas ao posicionamento do robô Pioneer 3DX e acompanhar sua movimentação.

Tendo em vista o baixo grau de formalismo para a avaliação de arquiteturas cognitivas no âmbito da robótica do desenvolvimento, a utilização dos experimentos de Berto (2020) para a validação da arquitetura proposta promoveram a verificação do aprendizado incremental em robôs e sua relação com as características sensório-motoras da Teoria de Jean Piaget (PIAGET, 1952).

6.2 Objetivos alcançados

A execução do projeto proporcionou a conquista dos seguintes objetivos:

- Modelagem e implementação de arquiteturas de software (classes e sub-classes) dos módulos cognitivos do modelo CONAIM, utilizando as ferramentas da computação cognitiva disponibilizadas no CST;
- Análise do relacionamento do desenvolvimento cognitivo com atenção, tomada de decisão e aprendizado procedimental no modelo CST+CONAIM;
- Implementação de um único mecanismo de aprendizado procedimental para demonstrar o processo incremental de aprendizado nos três primeiros subestágios sensório-motores da Teoria de Piaget;
- Implementação computacional de experimentos sensório-motores de **Rastreamento de Objetos**, análogos aos experimentos propostos por Berto (2020), compreendidos entre o 1^o e 3^o subestágio de desenvolvimento, utilizando como agente cognitivo a humanoide Marta;

6.3 Publicações

Além do texto desta dissertação, o seguinte artigo foi aceito para publicação em congresso internacional (*IEEE International Conference on Development and Learning (ICDL)*):

BERTO, L. M.; ROSSI, L. L.; ROHMER, E.; COSTA, P. D. P.; SIMOES, A. S.; GUDWIN, R. R.; COLOMBINI, E. L. Learning over the attentional space with mobile robots. *IEEE International Conference on Development and Learning (ICDL)*, 2020.

O seguinte artigo foi submetido para periódico internacional (*Cognitive Systems Research Journal*):

BERTO, L. M.; ROSSI, L. L.; ROHMER, E.; COSTA, P. D. P.; SIMOES, A. S.; GUDWIN, R. R.; COLOMBINI, E. L. An attentional model for cst. (*Submitted to*) *Cognitive Systems Research Journal*, 2020..

6.4 Trabalhos Futuros

Dentre as execuções que comporão a continuação deste trabalho, podem-se citar:

- Propor implementações computacionais baseadas nos demais experimentos sensório-motores delineados em Berto (2020);
- Promover a integração entre diferentes sensores para a realização destes experimentos;
- Analisar a estabilidade do Aprendizado Procedimental de acordo com o desvio padrão entre o número de ações executadas;
- Investigar o relacionamento entre atenção, aprendizado e a posição a cada instante dos atuadores físicos e virtuais;
- Análise do comportamento da agente com a movimentação de outros atuadores físicos;
- Analisar o relacionamento do desenvolvimento cognitivo com memórias e aprendizado no modelo CONAIM;
- Investigar a aplicabilidade da arquitetura em meio físico com a robô humanoide Marta.

REFERÊNCIAS

- ANDERSON, J.; LEBIERE, C. The newell test for a theory of mind. *Carnegie Mellon University*, 1998. Disponível em: <https://www.researchgate.net/publication/250037730_The_Newell_Test_for_a_Theory_of_Mind>. Acesso em: 2020-05-31.
- ARAÚJO, L. N. G. de; GUDWIN, R. R. *Desenvolvimento de um módulo atencional para o CST - Cognitive Systems Toolkit*. [S.l.], 2018. Disponível em: <<https://bv.fapesp.br/pt/bolsas/177535/desenvolvimento-de-um-modulo-atencional-para-o-cst-cognitive-systems-toolkit/>>.
- ARMSTRONG, K. H.; OGG, J.; A., S.-W.; A., W. *Early child development theories. Evidence-Based Interventions for Children with Challenging Behavior*. [S.l.]: Springer Science Business Media New York, 2014. ISBN 9781461478072.
- ARRABALES, R. *ConsScale - A Machine Consciousness Scale*. 2020. Disponível em: <<https://www.conscious-robots.com/consscale/index.html>>. Acesso em: 2020-06-13.
- ASADA, M.; MACDORMAN, K.; ISHIGURO, H.; KUNIYOSHI, Y. Synthetic neural modeling: the ‘darwin’ series of recognition automata. In: *Robotics and Autonomous Systems*. [s.n.], 2001. v. 37. Disponível em: <<https://ieeexplore.ieee.org/document/58327>>. Acesso em: 2020-06-13.
- BAARS, J. B. *A Cognitive Theory of Consciousness*. [S.l.]: Cambridge University Press, 1998. ISBN 0521427436.
- BARON-COHEN, S. Theory of mind and autism: A review. *International Review of Research in Mental Retardation*, p. 169–184, 2001. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0074775000800105>>. Acesso em: 2020-06-03.
- BEER, R. D. The Dynamics of Active Categorical Perception in an Evolved Model Agent. *Adaptive Behavior*, v. 11, n. 4, p. 209–243, 2003. ISSN 1059-7123, 1741-2633. Disponível em: <journals.sagepub.com/doi/10.1177/1059712303114001>. Acesso em: 2020-06-13.
- BELLAS, F.; DURO, R. J.; FAINA, A.; SOUTO, D. Multilevel Darwinist Brain (MDB): Artificial Evolution in a Cognitive Architecture for Real Robots. *IEEE Transactions on Autonomous Mental Development*, v. 2, n. 4, p. 340–354, 2010. ISSN 1943-0604, 1943-0612. Disponível em: <<http://ieeexplore.ieee.org/document/5599851>>.
- BENJAMIN, P.; LYONS, D.; LONSDALE, D. Designing a robot cognitive architecture with concurrency and active perception. In: *Proceedings of the AAAI Fall Symposium on the Intersection of Cognitive Science and Robotics*. [s.n.], 2004. Disponível em: <https://www.researchgate.net/publication/228783309_Designing_a_robot_cognitive_architecture_with_concurrency_and_active_perception>. Acesso em: 2020-06-13.
- BERLUCCHI, G.; MARZI, C. A. Neuropsychology of consciousness: Some history and a few new trends. *Frontiers in Psychology*, v. 10, p. 50, 2019. ISSN 1664-1078. Disponível em: <https://pdfs.semanticscholar.org/7bdc/53870598f99eb5d29e66b3315e7b7cc8572b.pdf?_ga=2.99375064.715380829.1591848230-931930848.1588334451>. Acesso em: 2020-06-06.
- BERTO, L. M. Exploring cognitive functions in robotics. *Institute of Computing of the University of Campinas (Master thesis)*, 2020.
- BERTO, L. M.; ROSSI, L. L.; ROHMER, E.; COSTA, P. D. P.; SIMOES, A. S.; GUDWIN, R. R.; COLOMBINI, E. L. An attentional model for cst. (*Submitted to*) *Cognitive Systems Research Journal*, 2020.
- BERTO, L. M.; ROSSI, L. L.; ROHMER, E.; COSTA, P. D. P.; SIMOES, A. S.; GUDWIN, R. R.; COLOMBINI, E. L. Learning over the attentional space with mobile robots. *IEEE International Conference on Development and Learning (ICDL)*, 2020.

- BOYD, D.; BEE, H. *A Criança em Crescimento*. [S.l.]: ARTMED, 2011. ISBN 9788536325408.
- BRAITENBERG, V. *Vehicles: Experiments in Synthetic Psychology*. [S.l.]: Cambridge MA: MIT Press, 1984. ISBN 9780262022088.
- BROOKS, R. A robust layered control system for a mobile robot. *A. I. Memo 864*, 1985. Disponível em: <https://people.csail.mit.edu/brooks/papers/AIM-864.pdf>. Acesso em: 2020-06-18.
- BROOKS, R. Intelligence without representation. *Artificial Intelligence*, v. 47, 1991. Disponível em: <https://people.csail.mit.edu/brooks/papers/representation.pdf>. Acesso em: 2020-05-28.
- BROOKS, R. From earwigs to humans. *Robotics and Autonomous Systems*, v. 20, 1997. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0921889096000644>. Acesso em: 2020-06-13.
- BROOKS, R. *Robot: The Future of Flesh and Machines*. [S.l.]: Penguin Books, 2003. ISBN 9780140297188.
- BROOKS, R.; STEIN, L. Building brains for bodies. *Autonomous Robots*, 1994. Disponível em: <http://people.csail.mit.edu/brooks/papers/brains.pdf>. Acesso em: 2020-05-28.
- CANGELOSI, A.; SCHLESINGER, M. *Developmental Robotics: From Babies to Robots*. The MIT Press, 2014. ISBN 9780262028011. Disponível em: <https://mitpress.mit.edu/books/developmental-robotics>. Acesso em: 2020-05-28.
- CAPITANIO, R.; SILVA, M.; GUDWIN, R. An introductory experiment with a conscious-based autonomous vehicle. In: *4th Workshop in Applied Robotics and Automation*. [s.n.], 2010. Disponível em: https://www.researchgate.net/publication/228955763_AN_INTRODUCTORY_EXPERIMENT_WITH_A_CONSCIOUS-BASED_AUTONOMOUS_VEHICLE. Acesso em: 2020-06-08.
- CASTRO, F. S.; LANDEIRA-FERNANDEZ, J. Alma, corpo e a antiga civilização grega: as primeiras observações do funcionamento cerebral e das atividades mentais. *psicologia: Reflexão crítica. scielo*, v. 24, p. 798—809, 2011. ISSN 0102-7972. Disponível em: <https://www.scielo.br/pdf/prc/v24n4/a21v24n4.pdf>. Acesso em: 2020-06-03.
- CHEC, D.; MARTIN, S. *Functional Movement Development Across the Life Span*. [S.l.]: Philadelphia: W.B. Saunders, 2002. ISBN 9781416049784.
- COLE, M.; COLE, S. *O desenvolvimento da criança e do adolescente*. [S.l.]: Porto Alegre: Artmed, 2003. ISBN 9788573079210.
- COLOMBINI, E. L. An attentional model for intelligent robotics agents. *Instituto Tecnológico de Aeronáutica, São José dos Campos (PhD Thesis)*, 2014.
- COLOMBINI, E. L.; SIMÕES, A. d. S.; RIBEIRO, C. H. C. An attentional model for autonomous mobile robots. *IEEE Systems Journal*, IEEE Systems Council, Washington, DC, USA, v. 11, n. 3, p. 1308–1319, 2017. ISSN 1932-8184. Disponível em: ieeexplore.ieee.org/document/7384702. Acesso em: 2020-06-10.
- COOK, J. L.; COOK, G. *Child development*. [S.l.]: London: Pearson education: Allyn Bacon., 2005. ISBN 9780205494064.
- DAUTENHAHN, K.; NUMAOKA, C. Studying robot social cognition within a developmental psychology framework. In: *3rd International Workshop on Advanced Mobile Robots*. Lausanne, Switzerland: [s.n.], 1999. Disponível em: <http://ieeexplore.ieee.org/document/827639>. Acesso em: 2020-06-12.
- EPSTEIN, S. L. Learning to play expertly: A tutorial on hoyle. In: FURNKRANZ, J.; KUBAT, M. (Ed.). *Machines that learn to play games*. Nova Science Publishers., 2001. p. 153—178. Disponível em: https://www.researchgate.net/publication/253938576_Learning_to_Play_Expertly_A_Tutorial_on_Hoyle. Acesso em: 2020-06-02.
- FONG, T.; NOURBAKHSH, I.; DAUTENHAHN, K. A survey of socially interactive robots. *Robotics and Autonomous Systems*, 2003. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S092188900200372X>. Acesso em: 2020-06-13.

FRANKLIN, S. A foundational architecture for artificial general intelligence. *Frontiers in Artificial Intelligence and Applications*, v. 157, p. 36–54, 06 2007. Disponível em: <https://www.researchgate.net/publication/234803101_A_Foundational_Architecture_for_Artificial_General_Intelligence>. Acesso em: 2020-06-13.

FRANKLIN, S. The mind according to lida – a brief account. Cognitive Computing Research Group, p. 20, 2012. Disponível em: <ccrg.cs.memphis.edu/tutorial/mindAccordingToLIDA/Brief-Account.pdf>. Acesso em: 2020-05-24.

FRANKLIN, S.; FAGHIHI, U.; MADL, T.; STRAIN, S.; DONG, D.; KUGELE, S.; SNAIDER, J.; AGRAWAL, P.; CHEN, S. A lida cognitive model tutorial. *Biologically Inspired Cognitive Architectures*, v. 16, p. 105–130, 2016. ISSN 2212-683X. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S2212683X16300196>>. Acesso em: 2020-05-24.

FRANKLIN, S.; FAGHIHI, U.; MCCALL, R. A computational model of attentional learning in a cognitive agent. *Biologically Inspired Cognitive Architectures*, v. 2, p. 25–36, 2012. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S2212683X12000321>>. Acesso em: 2020-05-24.

FUSTER, J. M. The cognit: A network model of cortical representation. *International Journal of Psychophysiology*, v. 60, p. 125–132, 2006.

GENNARO, R. Consciousness. *The Internet Encyclopedia of Philosophy*, 2020. ISSN 2161-0002. Disponível em: <<https://www.iep.utm.edu/consciou/>>.

GUDWIN, R. *Abordagens Paradigmáticas da Ciência Cognitiva*. 2014. Disponível em: <<http://www.dca.fee.unicamp.br/~gudwin/ftp/ia889/Aula08.pdf>>. Acesso em: 2020-06-13.

GUDWIN, R.; PARAENSE, A.; RAIZER, K.; PAULA, S. M. de; ROHMER, E.; FRÓES, E.; W., G.; CASTRO, E. An urban traffic controller using the meca cognitive architecture. *Biologically Inspired Cognitive Architectures*, v. 54, p. 26–41, 2017. Disponível em: <<https://www.sciencedirect.com/science/article/abs/pii/S2212683X18300859>>. Acesso em: 2020-06-16.

GUDWIN, R. R.; RAIZER, K.; PARAENSE, A. L. O.; PAULA, S. M. de; FIGUEIREDO, V. A. de; CASTRO, E. C. de; FRÓES, E.; GIBAUT, W. S. P. *CST - The Cognitive Systems Toolkit*. 2013. Disponível em: <cst.fee.unicamp.br>. Acesso em: 2020-06-13.

GUDWIN, R. R.; RAIZER, K.; PARAENSE, A. L. O.; PAULA, S. M. de; FIGUEIREDO, V. A. de; CASTRO, E. C. de; FRÓES, E.; GIBAUT, W. S. P. *CST - The Cognitive Systems Toolkit*. 2019. Disponível em: <<https://github.com/CST-Group/cst>>. Acesso em: 2020-06-13.

GUERIN, F.; KRUGER, N.; KRAFT, D. A Survey of the Ontogeny of Tool Use: From Sensorimotor Experience to Planning. *IEEE Transactions on Autonomous Mental Development*, v. 5, n. 1, p. 18–45, 2013. ISSN 1943-0604, 1943-0612. Disponível em: <ieeexplore.ieee.org/document/6248675/>. Acesso em: 2020-05-30.

HANFORD, S. D.; JANRATHITIKARN, O.; LONG, L. N. Control of mobile robots using the soar cognitive architecture. *JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION*, v. 6, 2009. Disponível em: <www.personal.psu.edu/~lnl/papers/jacic_2009.pdf>. Acesso em: 2020-05-29.

HARARI, Y. N. *Sapiens - Uma Breve História da Humanidade*. New York, NY, US: Harper, 2011.

HARRIS, C. L. Language and cognition. In: L., N. (Ed.). *Encyclopedia of cognitive science*. John Wiley and Sons, 2006. p. 1–6. Disponível em: <<https://onlinelibrary.wiley.com/doi/10.1002/0470018860.s00559>>. Acesso em: 2020-05-30.

HAYES-ROTH, B.; LALANDA, P.; MORIGNOT, P.; PFLEGER, K.; BALABANOVIC, M. Plans and behavior in intelligent agents. *KSL report n. 93*, 1993. Disponível em: <https://www.researchgate.net/publication/2427886_Plans_and_Behavior_in_Intelligent_Agents>. Acesso em: 2020-05-29.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. [S.l.]: Upper Saddle River, NJ, USA: Prentice Hall PT, 1998. v. 2. ISBN 0132733501.

- HOFSTADER, D. R.; MITCHELL, M. The copycat project: A model of mental fluidity and analogy-making. p. 31–112, 1995. Disponível em: https://www.researchgate.net/publication/243649607_The_copycat_project_A_model_of_mental_fluidity_and_analogy-making. Acesso em: 2020-05-24.
- JAMES, W. The principles of psychology. *New York: Henry Holt and Company.*, 1890. Disponível em: http://library.manipaldubai.com/DL/the_principles_of_psychology_vol_I.pdf. Acesso em: 2020-06-03.
- KIERAS, D. E. Epic architecture principles of operation. *University of Michigan.*, 2004. Disponível em: <https://web.eecs.umich.edu/~kieras/docs/EPICtutorial2004/EPICPrinOp.pdf>. Acesso em: 2020-06-02.
- KIRK, J. R.; LAIRD, J. E. Learning hierarchical symbolic representations to support interactive task learning and knowledge transfer. In: *Twenty-Eighth International Joint Conference on Artificial Intelligence*. [s.n.], 2019. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.677.223>. Acesso em: 2020-05-29.
- KOCH, C.; ULLMAN, S. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, v. 4, 1985. Disponível em: <https://cseweb.ucsd.edu/classes/fa09/cse258a/papers/koch-ullman-1985.pdf>. Acesso em: 2020-06-02.
- KOCH, S.; GREENFIELD, S. How does consciousness happen? *SCIENTIFIC AMERICAN*, 2007. Disponível em: https://www.researchgate.net/publication/5917959_How_Does_Consciousness_Happen. Acesso em: 2020-06-08.
- KOTSERUBA, I.; TSOTSOS, J. 40 years of cognitive architectures: core cognitive abilities and practical applications. *Artificial Intelligence Review*, v. 53, 2020. Disponível em: link.springer.com/article/10.1007/s10462-018-9646-y. Acesso em: 2020-06-12.
- KOTSERUBA, I.; TSOTSOS, J. Action selection. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/action_selection.html.
- KOTSERUBA, I.; TSOTSOS, J. Applications. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/applications.html.
- KOTSERUBA, I.; TSOTSOS, J. Bibliographies. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/bib_html/index.html.
- KOTSERUBA, I.; TSOTSOS, J. Learning. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/learning.html.
- KOTSERUBA, I.; TSOTSOS, J. Memory. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/memory.html.
- KOTSERUBA, I.; TSOTSOS, J. Perception. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/perception.html.
- KOTSERUBA, I.; TSOTSOS, J. Source code / software link. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/source_code.html.
- KOTSERUBA, I.; TSOTSOS, J. Visual attention. In: *40 years of cognitive architectures: core cognitive abilities and practical applications*. Artificial Intelligence Review, 2020. Disponível em: http://jtl.lassonde.yorku.ca/project/cognitive_architectures_survey/visual_attention.html.
- KRICHMAR, J.; NITZ, D.; GALLY, J.; EDELMAN, M. Characterizing functional hippocampal pathways in a brain-based device as it solves a spatial memory task. *The Neurosciences Institute, John Jay Hopkins Drive, San Diego*, 2004. Disponível em: https://www.researchgate.net/publication/8048446_Characterizing_functional_hippocampal_pathways_in_a_brain-based_device_as_it_solves_a_spatial_memory_task. Acesso em: 2020-05-26.

LAIRD, J. *The Soar cognitive architecture*. Cambridge, Mass. ; London, England: MIT Press, 2012. ISBN 9780262122962.

LAIRD, J. E. Extending the soar cognitive architecture. In: *Proceedings of the First AGI Conference - Frontiers in Artificial Intelligence and Applications*. Amsterdam, Netherlands: Artificial General Intelligence by P. Wang, B. Goertzel, and S. Franklin, 2008. p. 224—235. Disponível em: https://www.researchgate.net/publication/221328941_Extending_the_Soar_Cognitive_Architecture. Acesso em: 2020-06-11.

LAIRD, J. E.; CONGDON, C. B.; ASSANIE, M.; DERBINSKY, N.; XU, J. *Manual Soar User Version 9.6.0*. 2017. Disponível em: <https://soar.eecs.umich.edu/downloads/SoarManual.pdf>. Acesso em: 2020-06-13.

LAIRD, J. E.; KINKADE, K. R.; MOHAN, S.; XU, J. Z. Cognitive robotics using the soar cognitive architecture. *Cognitive Robotics AAAI Technical Report*, 2012. Disponível em: <https://www.semanticscholar.org/paper/Cognitive-Robotics-Using-the-Soar-Cognitive-Laird-Kinkade/1020e432b927878635407301d85f5569add990a0>. Acesso em: 2020-05-29.

LANGLEY, P.; CHOI, D.; TRIVEDI, N. Icarus users manual. *Technical Report, Computational Learning Laboratory, Center for the Study of Language and Information*, 2008. Disponível em: <https://pdfs.semanticscholar.org/2ec0/d0fa640ba6d66c6430c90a8ee521e75f4e96.pdf>. Acesso em: 2020-06-13.

LEFRANÇOIS, G. R. *Teorias da Aprendizagem*. [S.l.]: Cengage Learning, 2008. ISBN 9788522106226.

LICATO, J.; MARTON, N.; DONG, B.; SUN, R.; BRINGSJORD, S. Modeling the creation and development of cause-effect pairs for explanation generation in a cognitive architecture. In: *International Workshop on Artificial Intelligence and Cognition*. Torino, Italy: [s.n.], 2015. p. 29–39.

LUNGARELLA, M.; BERTHOUE, L. On the interplay between morphological, neural and environmental dynamics: a robotic case-study. *Adaptive Behavior*, v. 10, 2002. Disponível em: https://www.researchgate.net/publication/225304134_On_the_Interplay_Between_Morphological_Neural_and_Environmental_Dynamics_A_Robotic_Case_Study. Acesso em: 2020-05-26.

LUNGARELLA, M.; SANDINI, G.; PFEIFER, R. Developmental robotics: A survey. *Connection Science*, v. 15, n. 4, p. 151–190, 2003. Disponível em: https://www.researchgate.net/publication/220233671_Developmental_robotics_A_survey. Acesso em: 2020-06-12.

MADL, T.; FRANKLIN, S.; CHEN, K.; TRAPPL, R.; MONTALDI, D. Exploring the structure of spatial representations. *PLOS ONE*, 2016. Disponível em: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0157343>. Acesso em: 2020-06-13.

MEEDEN, L. A.; BLANK, D. S. Introduction to developmental robotics. *Connection Science*, v. 18, n. 2, p. 93–96, 2006. ISSN 0954-0091, 1360-0494. Disponível em: www.tandfonline.com/doi/abs/10.1080/09540090600806631. Acesso em: 2020-06-14.

MELTZOFF, A.; MOORE, M. Imitation of facial and manual gestures by human neonates. *Science*, v. 198, 1977. Disponível em: <https://pubmed.ncbi.nlm.nih.gov/17741897/>. Acesso em: 2020-06-12.

MICHALEC, D. Bayley scales of infant development: Third edition. In: _____. *Encyclopedia of Child Behavior and Development*. Boston, MA: Springer US, 2011. p. 215–215. ISBN 978-0-387-79061-9. Disponível em: https://doi.org/10.1007/978-0-387-79061-9_295.

MININGER, A.; LAIRD, J. Interactively learning strategies for handling references to un-seen or unknown objects. *Advanced Cognitive Systems*, 2016. Disponível em: <https://www.semanticscholar.org/paper/Interactively-Learning-Strategies-for-Handling-to-Mininger-Laird/40b256b0110ca660464473f18394c8fba757d1b8>. Acesso em: 2020-06-11.

MOHAN, S.; LAIRD, J. A case study of knowledge integration across multiple memories in soar. *Biologically Inspired Cognitive Architectures*, p. 93—99, 2014. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S2212683X14000164>. Acesso em: 2020-06-11.

NYAMSUREN, E.; TAATGEN, N. A. Pre-attentive and attentive vision module. *Cogn. Syst. Res.*, p. 211—216, 2013. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S1389041712000630>>. Acesso em: 2020-05-31.

OUDEYER, P.-Y.; KACHERGIS, G.; SCHUELLER, W. *Computational and Robotic Models of Early Language Development: A Review*. [s.n.], 2019. Disponível em: <<https://arxiv.org/ftp/arxiv/papers/1903/1903.10246.pdf>>. Acesso em: 2020-05-28.

PAISNER, M.; COX, M. T.; MAYNORD, M.; PERLIS, D. Goal-driven autonomy for cognitive systems. *Cognitive science society*, 2013. Disponível em: <<https://www.semanticscholar.org/paper/Goal-Driven-Autonomy-for-Cognitive-Systems-Paisner-Cox/16c2dd7428116177c68b0747b64bb16b13ee5e4a>>. Acesso em: 2020-06-02.

PAPALIA, D. E.; FELDMAN, R. D. *Desenvolvimento Humano*. [S.l.]: Mcgraw Hill - Artmed, 2013. v. 12. ISBN 9788580552171.

PARAENSE, A. L. O.; RAIZER, K.; GUDWIN, R. R. A machine consciousness approach to urban traffic control. *Biologically Inspired Cognitive Architectures*, v. 15, p. 61–73, 2016. ISSN 2212-683X. Disponível em: <www.sciencedirect.com/science/article/pii/S2212683X15000614>. Acesso em: 2020-06-10.

PARAENSE, A. L. O.; RAIZER, K.; PAULA, S. M. de; GUDWIN, R. R.; ROHMER, E. The cognitive systems toolkit and the cst reference cognitive architecture. *Biologically Inspired Cognitive Architectures*, v. 17, p. 32–48, 2016. ISSN 2212-683X. Disponível em: <www.sciencedirect.com/science/article/pii/S2212683X1630038X>. Acesso em: 2020-06-13.

PFEIFER, R. Design principles for autonomous agents: A case study of classification. *Artificial Life and Robotics*, 1997. Disponível em: <<https://link.springer.com/article/10.1007/BF02471112>>. Acesso em: 2020-05-26.

PFEIFER, R.; SCHEIER, C. From perception to action: The right direction? *IEEE Computer Society Press*, 1994. Disponível em: <https://www.researchgate.net/publication/3722874_From_Perception_to_Action_The_Right_Direction>. Acesso em: 2020-05-26.

PIAGET, J. *The origins of intelligence in children*. New York, NY, US: W Norton & Co, 1952. (The origins of intelligence in children.).

PIAGET, J. *Science of Education and the Psychology of the Child*. [S.l.]: Penguin Books, 1971. ISBN 9780670003112.

PIAGET, J. *Seis estudos da Psicologia*. [S.l.]: Forense Universitária, 1999. ISBN 8521802463.

POSNER, M. *Orienting of attention*. *The Quarterly Journal of Experimental Psychology*. [s.n.], 1980. v. 32. Disponível em: <https://www.researchgate.net/publication/15830678_Orienting_of_Attention>. Acesso em: 2020-05-24.

PRATAMA, F.; MASTROGIOVANNI, F.; CHONG, N. Y. An integrated epigenetic robot architecture via context-influenced long-term memory. In: *4th International Conference on Development and Learning and on Epigenetic Robotics*. Genoa, Italy: IEEE, 2014. p. 68–74. ISBN 9781479975402. Disponível em: <<https://ieeexplore.ieee.org/document/6982956/>>. Acesso em: 2020-05-23.

RAIZER, K. *EXECUTIVE FUNCTIONS FOR LEARNING AND DECISION-MAKING IN A BIO-INSPIRED COGNITIVE ARCHITECTURE*. Tese (Doctor in Electrical Engineering) — University of Campinas, Campinas. São Paulo. Brasil., 2015.

RAIZER, K.; PARAENSE, A. L. O.; GUDWIN, R. R. A cognitive neuroscience-inspired codelet-based cognitive architecture for the control of artificial creatures with incremental levels of machine consciousness. In: *Proceedings of a symposium at the AISB11 Convention. Machine Consciousness 2011: Self, Integration and Explanation*. York, United Kingdom: UK Society for the Study of Artificial Intelligence and Simulation of Behaviour, 2011. p. 43–50. Disponível em: <[http://www.dca.fee.unicamp.br/~klaus/PUB/\[Raizer_et_al_2011\]_A_Cognitive_Neuroscience-inspired_Codelet-based_Cognitive_Architecture.pdf](http://www.dca.fee.unicamp.br/~klaus/PUB/[Raizer_et_al_2011]_A_Cognitive_Neuroscience-inspired_Codelet-based_Cognitive_Architecture.pdf)>. Acesso em: 2020-06-10.

RAIZER, K.; PARAENSE, A. L. O.; GUDWIN, R. R. A cognitive architecture with incremental levels of machine consciousness inspired by cognitive neuroscience. *International Journal of Machine Consciousness*, World Scientific, v. 04, n. 02, p. 335–352, 2012. Disponível em: www.researchgate.net/publication/263906381_A_cognitive_architecture_with_incremental_levels_of_machine_consciousness_inspired_by_cognitive_neuroscience. Acesso em: 2020-06-10.

REGATTIERI, C.; COLOMBINI, E. Aprendendo sobre o espaço atencional no cst - the cognitive systems toolkit. *Institute of Computing of the University of Campinas (Bachelor thesis)*, 2018. Disponível em: <https://ic.unicamp.br/~reltech/PFG/2018/PFG-18-31.pdf>. Acesso em: 2020-06-16.

REGGIA, J. The rise of machine consciousness: studying consciousness with computational models. *Neural Networks*, v. 44, 2013. ISSN 112-131. Disponível em: www.sciencedirect.com/science/article/abs/pii/S0893608013000968. Acesso em: 2020-05-26.

ROCHAT, P.; STRIANO, T. Perceived self in infancy. *Infant Behavior Development*, 2000. Disponível em: <https://www.semanticscholar.org/paper/Perceived-self-in-infancy-Rochat-Striano/f86b91858fb285b6ff22bea29d656c65882ed67>. Acesso em: 2020-06-13.

ROHMER, E.; FREESE, M. *Robot Simulator CoppeliaSim*. 2019. Disponível em: <https://www.coppeliarobotics.com/>. Acesso em: 2020-05-30.

ROHMER, E.; SINGH, S. P. N.; FREESE, M. V-rep: a versatile and scalable robot simulation framework. In: *Proc. of The International Conference on Intelligent Robots and Systems (IROS)*. [s.n.], 2013. Disponível em: <https://ieeexplore.ieee.org/document/6696520>. Acesso em: 2020-06-02.

RUSSEL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. [S.l.]: Alan Apt, 1995.

SAMSONOVICH, A. *BICA Society: Comparative Table of Cognitive Architectures*. 2010. Disponível em: <https://bicasociety.org/cogarch/architectures.htm>. Acesso em: 2020-06-13.

SAMSONOVICH, A. Toward a unified catalog of implemented cognitive architectures. In: *First Annual Meeting of the BICA Society*. Amsterdam, The Netherlands: BICA Society, IOS Press, 2010. p. 195—244. Disponível em: https://www.researchgate.net/publication/221313271_Toward_a_Unified_Catalog_of_Implemented_Cognitive_Architectures. Acesso em: 2020-06-13.

SAUTER, E.; AZEVEDO, F.; STRAUCH, I. A função de heaviside. *Universidade Federal do Rio Grande do Sul*, 2019.

SCASSELLATI, B. Foundations for a theory of mind for a humanoid robot. *MIT Department of Electrical Engineering and Computer Science (PhD thesis)*, 2001. Disponível em: <https://groups.csail.mit.edu/lbr/hrg/2001/scassellati-phd.pdf>. Acesso em: 2020-06-13.

SEARLE, J. R. The problem of consciousness. *Consciousness Cogn.*, 1993. Disponível em: www.faculty.umb.edu/adam_beresford/courses/phil_100_11/reading_problem_of_consciousness.pdf. Acesso em: 2020-06-09.

SIMÕES, A. d. S.; COLOMBINI, E. L.; RIBEIRO, C. H. C. CONAIM: A Conscious Attention-Based Integrated Model for Human-Like Robots. *IEEE Systems Journal*, v. 11, n. 3, p. 1296–1307, 2017. ISSN 1932-8184, 1937-9234, 2373-7816. Disponível em: ieeexplore.ieee.org/document/7383269. Acesso em: 2020-06-13.

SIMÕES, A. d. S.; COLOMBINI, E. L.; RIBEIRO, C. H. C. Conaim: A conscious attention-based integrated model for human-like robots. *IEEE Systems Journal*, IEEE Systems Council, Washington, DC, USA, 2016. ISSN 1932-8184. Disponível em: <https://ieeexplore.ieee.org/document/7383269>. Acesso em: 2020-06-13.

SIMÕES, A. S. Um modelo cognitivo baseado na atenção para consciência em robôs inteligentes. *Universidade Estadual Paulista (Tese de Livre-Docência)*, 2015.

SNAIDER, J.; MCCALL, R.; STRAIN, S.; FRANKLIN, S. *The LIDA Tutorial Version 1.0*. 2012. Disponível em: <http://ccrg.cs.memphis.edu/assets/framework/The-LIDA-Tutorial.pdf>. Acesso em: 2020-06-13.

SQUIRE, L. R. Declarative and nondeclarative memory: multiple brain systems supporting learning and memory. *J Cogn Neurosci.*, v. 4, p. 232—234, 1992. Disponível em: <researchgate.net/publication/8251982_Memory_Systems_of_the_Brain_A_Brief_History_and_Current_Perspective>. Acesso em: 2021-08-31.

SQUIRE, L. R. Memory systems of the brain: a brief history and current perspective. *Neurobiology of Learning and Memory*, v. 82, p. 171—177, 2004. Disponível em: <researchgate.net/publication/8251982_Memory_Systems_of_the_Brain_A_Brief_History_and_Current_Perspective>. Acesso em: 2020-05-27.

STEWART, T. C.; BLOUW, P.; ELIASMITH, C. Explorations in distributed recurrent biological parsing. *International Conference on Cognitive Modelling.*, 2015. Disponível em: <http://www.cognitive-modeling.com/proceedings/ICCM2015_proceedings.pdf>. Acesso em: 2020-06-02.

STOYTICHEV, A. Some basic principles of developmental robotics. *IEEE TRANSACTIONS ON AUTONOMOUS MENTAL DEVELOPMENT*, v. 1, 2009. Disponível em: <<https://ieeexplore.ieee.org/document/5200465?arnumber=5200465>>. Acesso em: 2020-06-13.

SUN, R. Accounting for the computational basis of consciousness. a connetionist approach. *Consciousness and cognition*, v. 8, p. 529—565, 1999. Disponível em: <<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.42.2681&rep=rep1&type=pdf>>. Acesso em: 2020-05-25.

SUN, R. Desiderata for cognitive architectures. *Philosophical Psychology*, v. 17, n. 3, p. 341—373, 2004. ISSN 0951-5089, 1465-394X. Disponível em: <<https://www.semanticscholar.org/paper/Desiderata-for-cognitive-architectures-Sun/cf3eeb7edb83340ddcf234b353ca543f98373394>>. Acesso em: 2020-05-28.

SUN, R.; FRANKLIN, S. Computational models of consciousness. *Cambridge handbook of consciousness*. Cambridge: Cambridge University Press., p. 151—174, 2007. Disponível em: <https://www.researchgate.net/publication/251758834_Computational_Models_of_Consciousness_A_Taxonomy_and_Some_Examples>. Acesso em: 2020-06-10.

SUN, R.; FRANKLIN, S. *Anatomy of the Mind: Exploring Psychological Mechanisms and Processes with the Clarion Cognitive Architecture*. [S.l.]: Oxford University Press Inc., 2016. ISBN 9780199794560.

SUN, R.; WILSON, N.; LYNCH, M. *CLARION Cognitive Architecture Project*. 2019. Disponível em: <<http://www.clarioncognitivearchitecture.com/>>. Acesso em: 2020-06-13.

SUTTON, R.; BARTO, A. *Reinforcement Learning: An Introduction*. [S.l.]: MIT Press. Cambridge, Massachussets., 1998. ISBN 9780262193986.

THAGARD, P. Cognitive science. In: ZALTA, E. N. (Ed.). *The Stanford Encyclopedia of Philosophy*. Spring 2019. Center for the Study of Language and Information, Stanford University, 2019. Disponível em: <<https://plato.stanford.edu/archives/win2018/entries/cognitive-science/>>. Acesso em: 2020-06-05.

THELEN, E.; SMITH, L. *A Dynamic Systems Approach to the Development of Cognition and Action*. [S.l.]: Cambridge MA: MIT Press. A Bradford Book, 1994. ISBN 9780262284875.

THOMAS, J. Current status of consciousness research from the neuroscience perspective. *Acta Scientific Neurology*, v. 2, p. 38—44, 2019. Disponível em: <https://www.researchgate.net/publication/330873739_Current_Status_of_Consciousness_Research_from_the_Neuroscience_Perspective>. Acesso em: 2020-06-07.

THOMSON, R.; BENNATI, S.; LEBIERE, C. Extending the influence of contextual information in act-r using buffer decay. In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. [s.n.], 2014. p. 6. Disponível em: <www.researchgate.net/publication/268520256_Extending_the_Influence_of_Contextual_Information_in_ACT-R_using_Buffer_Decay>. Acesso em: 2020-06-02.

TOMAZELA, R.; COLOMBINI, E. Uma abordagem combinada de planejamento baseado em modelo e aprendizado por reforço para locomoção bípede. *Institute of Computing of the University of Campinas (Bachelor thesis)*, 2019. Disponível em: <http://repositorio.unicamp.br/bitstream/REPOSIP/335166/1/Tomazela_RafaelMariottini_M.pdf>. Acesso em: 2020-06-16.

- TONONI, G. An information integration theory of consciousness. *BMC Neuroscience*, v. 5, 2016. Disponível em: <https://www.researchgate.net/publication/8198745_An_Information_Integration_Theory_of_Consciousness>. Acesso em: 2020-06-08.
- TSOTSOS, J. K. *A computational perspective on visual attention*. [S.l.]: MIT Press Cambridge, 2011. ISBN 9780262015417.
- TULVING, E. How many memory systems are there? *American Psychologist*, v. 60, p. 385–398, 1985. Disponível em: <<https://psycnet.apa.org/record/1986-08252-001>>. Acesso em: 2020-06-02.
- VERNON, D.; HOFSTEN, C. von; FADIGA, L. Desiderata for developmental cognitive architectures. *Biologically Inspired Cognitive Architectures*, v. 18, p. 116–127, 2016. ISSN 2212683X. Disponível em: <https://www.researchgate.net/publication/309333693_Desiderata_for_developmental_cognitive_architectures>. Acesso em: 2020-05-23.
- VERNON, D.; METTA, G.; SANDINI, G. The icub cognitive architecture: Interactive development in a humanoid robot. *IEEE Conf. Develop. Learn.*, 2007. Disponível em: <<https://ieeexplore.ieee.org/document/4354038/>>. Acesso em: 2020-06-02.
- VYGOTSKY, L. *Thought and Language*. [S.l.]: Cambridge MA: MIT Press, 1986. Original: 1934. ISBN 139780262220293.
- WADSWORTH, B. J. *Inteligência e afetividade da criança na teoria de Piaget*. [S.l.]: Pioneira (São Paulo), 1997. ISBN 9780801307737.
- WANG, Y.; MILLER, J.; LIU, T. Suppression effects in feature-based attention. *Journal of Vision*, 2015. Disponível em: <<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4419381/>>. Acesso em: 2020-06-02.
- WATKINS, C.; DAYAN, P. Q-learning. *Machine Learning*, v. 8, p. 279–292, 1992. Disponível em: <<https://link.springer.com/content/pdf/10.1007/BF00992698.pdf>>. Acesso em: 2020-05-29.
- WENG, J.; LEE, Y.-B.; EVANS, C. H. The developmental approach to multimedia speech learning. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1999. Disponível em: <<https://www.computer.org/csdl/proceedings-article/icassp/1999/00757495/12OmNBpmDOI>>. Acesso em: 2020-06-02.
- WENG, M.; JOHANNES, C. J. Preparation of glycosyltriazenes. *Adaptation in Artificial and Biological Systems*, Bristol UK: Society for the Study of Artificial Intelligence and the Simulation of Behaviour, 2006.
- WOOD, K. C.; SMITH, H.; GROSSNIKLAUS, D. Piaget's stages of cognitive development. *Department of Educational Psychology and Instructional Technology, University of Georgia*, 2011. Disponível em: <<https://resources.saylor.org/wwwresources/archived/site/wp-content/uploads/2011/07/psych406-5.3.2.pdf>>. Acesso em: 2020-06-13.
- WOOLFOLK, A. E.; WINNE, P. H.; PERRY, N. E. *Educational psychology*. [S.l.]: Pearson Education Canada Inc: Allyn Bacon, Inc Needham Height, MA, 2010. ISBN 9783621279178.
- WUNDT, W. *Outlines of Psychology*. Leipzig: W. Engleman., 1897. Disponível em: <library.manipaldubai.com/DL/outlines_of_psychology.pdf>. Acesso em: 2020-06-03.

Experimentos sensório-motores

Experimentos computacionais propostos por [Berto \(2020\)](#) para a incorporação de comportamentos humanos em robôs e para análise do desenvolvimento cognitivo artificial. Os experimentos são apresentados em uma sequência de subetapas, com cada novo subestágio se relacionando de forma incremental com os anteriores.

Rastrear Objetos com Sensor de Visão			
Subestágio	Tarefa	Comentários	Resultado esperado
1	Posicionar o robô em frente a um objeto à 25cm	Objeto dentro do campo de visão do robô, em posição fixa e de cor primária. Acuidade visual: 20/400. Campo de Visão pequeno	Olhar para o objeto
1	Posicionar o robô em frente a um objeto à 25cm	Objeto dentro do campo de visão do robô, se movimentando lentamente e de cor primária. Acuidade visual: 20/400. Campo de Visão pequeno	Acompanhar o objeto com o olhar
1	Posicionar o robô em frente a um objeto à 25cm	Objeto dentro do campo de visão do robô, se movimentando rapidamente e de cor primária. Acuidade visual: 20/400. Campo de Visão pequeno	Não acompanha o objeto com o olhar
2	Posicionar o robô em frente a um objeto à 80cm	Objeto dentro do campo de visão do robô, se movimentando em uma velocidade média e de cor primária. Acuidade visual: 20/400. Aumentar Campo de Visão	Acompanhar o objeto com o olhar
2	Posicionar o robô próximo a um objeto à 60cm	Objeto fora do campo de visão do robô, em posição fixa e de cor primária. Acuidade visual: 20/100. Aumentar Campo de Visão	Não olha para o objeto
2	Posicionar o robô próximo a um objeto à 60cm	Objeto de cor primária entra, com uma velocidade média, no campo de visão do robô. Acuidade visual: 20/100. Aumentar Campo de Visão	Acompanha o objeto enquanto este está em seu campo de visão
2	Posicionar o robô próximo a um objeto à 60cm	Objeto de cor primária sai, com uma velocidade média, do campo de visão do robô. Acuidade visual: 20/100. Aumentar Campo de Visão	Acompanha o objeto enquanto este está em seu campo de visão
3	Posicionar o robô em frente a um objeto à 150cm	Objeto dentro do campo de visão do robô, se movimentando em uma velocidade média e de cor primária. Acuidade visual: 20/20. Aumentar Campo de Visão	Acompanhar o objeto com o olhar
3	Posicionar o robô em frente a um objeto à 150cm	Objeto de cor primária entra, com uma velocidade média, no campo de visão do robô. Acuidade visual: 20/20. Aumentar Campo de Visão	Acompanhar o objeto com o olhar e virar a cabeça para continuar olhando para o objeto
3	Posicionar o robô em frente a dois objetos, com diferentes distâncias	Objetos de cor primária e mesmos tamanhos, um mais próximo do robô que o outro.	Notar que o objeto que parece menor está mais distante.
3	Posicionar o robô em frente a dois objetos, com diferentes distâncias, um sobrepondo o outro	Objetos de cor primária e mesmos tamanhos, um mais próximo do robô que o outro e um cobrindo parte do outro.	Notar que o objeto que está parcialmente oculto está mais distante.

Tabela 3 – Experimentos relacionados com a habilidade de rastrear objetos. Fonte: ([BERTO, 2020](#)).

Valores absolutos dos gráficos de número de ações executadas para cada possível ação

Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios durante o treinamento do Aprendizado Procedimental no 1º subestágio e durante a validação com os experimentos de [Berto \(2020\)](#). a_{M1} (manter o foco); a_{M2} (*NeckYaw* p/ esquerda); a_{M3} (*NeckYaw* p/ direita); a_{M4} (*HeadPitch* p/ baixo); a_{M5} (*HeadPitch* p/ cima); a_{M6} (Fóvea p/ posição 0); a_{M7} (Fóvea p/ posição 1); a_{M8} (Fóvea p/ posição 2); a_{M9} (Fóvea p/ posição 3); a_{M10} (Fóvea p/ posição 4).

(1º Subestágio) Aprendizado Procedimental										
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10
1 - 10	2,5 ±5,5	5 ±3,2	2 ±3	2 ±3	2,8 ±2,7	7,5 ±5	1 ±1	0 ±0	0 ±0	0 ±0
11 - 20	1 ±2	1 ±2	4,5 ±3,7	7 ±1	1,9 ±1,3	11,9 ±14	3,6 ±2,1	0 ±1	0 ±0	1 ±0,5
21 - 30	3 ±2,5	4 ±3	2,4 ±2	5 ±2,3	1,9 ±1,3	6,4 ±4,2	2,8 ±2,2	2 ±2,7	2 ±3	1 ±1,5
31 - 40	5 ±3	7,6 ±5,4	5,5 ±3,5	5 ±3	8,1 ±2,2	17,3 ±9,5	7,4 ±4,6	2 ±3,2	4 ±2	7,8 ±2,9
41 - 50	5 ±2	5,8 ±4,3	7,6 ±3,2	8 ±3	4,5 ±2,7	8,8 ±5	9,3 ±8	5,3 ±3,2	1 ±1	2,9 ±1,9
51 - 60	5 ±3,2	2,3 ±2,1	5 ±4,7	5 ±1	3,6 ±2,2	4,6 ±2,6	3,7 ±3,2	2 ±1,2	0 ±0	2,9 ±2,5
61 - 70	18 ±8,6	11,7 ±5,2	9,7 ±5,7	10,2 ±4,3	9,9 ±3,7	14,4 ±8	12,8 ±7,3	5,2 ±3,2	3,1 ±2	5,9 ±4,5
71 - 80	9,6 ±5,7	3,3 ±2	9,8 ±5	9,7 ±4	8,2 ±5,6	9,2 ±8	10,2 ±6,4	4,3 ±3,1	4 ±3,4	2,9 ±2
81 - 90	12,5 ±7,1	14,6 ±6,7	13 ±7	12,7 ±4,8	15,4 ±4,6	16,4 ±5,5	16,5 ±6,4	5,3 ±2,2	6 ±4,1	5 ±3
91 - 100	12,5 ±5,3	9,8 ±4,8	11 ±4	8,7 ±5,3	8,2 ±3,3	8,3 ±4	9,3 ±3	4,2 ±1,7	2 ±1,4	6,8 ±3,4
101 - 110	14,3 ±5,2	9,7 ±4,3	11 ±6	12 ±6,6	11,7 ±5,6	14,6 ±7,3	15,7 ±9,1	5,3 ±2,8	6 ±3	5 ±2,4
111 - 120	11,5 ±4,9	10,4 ±5,2	14 ±7	10,2 ±5,4	12,7 ±4,5	9,2 ±5,4	8,2 ±4,6	4,4 ±3,2	6 ±4,5	4,9 ±2,5
121 - 130	11,5 ±4,5	12,7 ±7,3	11,5 ±6,2	13,8 ±5	13,6 ±5,3	11,8 ±5,4	13,3 ±9,1	8,4 ±3,6	5,1 ±2,5	5,9 ±1,5
131 - 140	7,8 ±3,1	7,8 ±3	5,3 ±2,3	6,3 ±3,5	13,7 ±2,2	11 ±3,2	10,2 ±4,1	3,3 ±1,6	4 ±1,5	4 ±1,5
141 - 150	14,2 ±6,2	10,8 ±5	14 ±6,3	16 ±5,7	15,4 ±6,6	11,5 ±8,5	13,8 ±6,7	7,5 ±2,6	10 ±6,1	6 ±1,5
151 - 160	9,7 ±4,3	9,8 ±5	10,4 ±4,3	10,2 ±1,7	11,7 ±3,6	9,2 ±2,5	9,2 ±2,2	5,4 ±1,9	4 ±1,4	4 ±1,5
161 - 170	11,6 ±5,6	9,8 ±5	9,5 ±3,5	8,2 ±3,8	8,2 ±4,3	10,1 ±4	8,4 ±4	6,5 ±3,2	5,2 ±1	3 ±1,3
171 - 180	10,9 ±5,6	14,7 ±5	11 ±5	13,2 ±3	11,9 ±4,2	11 ±2,6	10,2 ±2,7	6,5 ±2,6	6,5 ±2,6	5,2 ±1
181 - 190	14,2 ±4,5	14,7 ±5,8	14,8 ±6	13,3 ±3,4	12,8 ±5,8	15,4 ±6,4	13,8 ±6,4	7,5 ±2,7	6,9 ±3,4	5,9 ±2,5
191 - 200	8 ±3,3	9,4 ±4,2	9 ±4,6	6,6 ±2,1	9,1 ±1,8	9,2 ±2,7	8,3 ±2,4	4,4 ±1	4 ±1	6,1 ±3,4
(1º Subestágio) Exp. Berto A - Objeto em posição fixa e de cor primária										
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10
1 - 10	9 ±4,3	10,9 ±7,2	13 ±7	11,9 ±5	8 ±4,5	9,9 ±5,5	10 ±5,6	4,2 ±2,6	5 ±3	6 ±3
11 - 20	13,9 ±8,1	16,7 ±8,7	16 ±10	15,8 ±8,5	12,9 ±8,4	17,8 ±11,4	15,3 ±8,6	11 ±4,7	7 ±4,1	6 ±3,5
21 - 30	20,8 ±8	23,8 ±9,8	24,6 ±10,4	21,8 ±9,5	19,9 ±8,7	22,8 ±8,5	22,2 ±10,1	12 ±5,2	12,2 ±5,2	10 ±4,4
31 - 40	11,3 ±3,9	13,1 ±5	11 ±4,1	11,7 ±3,6	9 ±4,5	14,9 ±5	11,1 ±5	11 ±4,5	5 ±2,7	2,9 ±1,4
41 - 50	16,8 ±7,2	7,3 ±4,3	21 ±9,8	19,9 ±8,9	18 ±10,2	18,9 ±9,3	15 ±8	8 ±3,5	9,1 ±5	7 ±3,5
51 - 60	15,8 ±7	18 ±7,4	16 ±4,4	17,8 ±6,6	17 ±7,6	19,8 ±7,9	15 ±5,5	8 ±3	7,2 ±4,1	7,9 ±3,5
61 - 70	12,1 ±4,6	13,9 ±3,2	13,2 ±4,2	11,9 ±2,6	9 ±3	10 ±3,1	13 ±4	5 ±2	3,1 ±2	5 ±2
71 - 80	10,7 ±4,2	11,8 ±4	11,1 ±4,6	12,9 ±4,2	11 ±4,5	13,9 ±5	11,2 ±6	4 ±2	7 ±3	5 ±2
81 - 90	16,8 ±8,4	17,7 ±9,2	19 ±10	16,9 ±8,4	20 ±6,4	19,8 ±10,8	19,3 ±9	8 ±5,5	9 ±4,6	8,9 ±4,9
91 - 100	15,8 ±3,2	19,8 ±4,2	19 ±4,6	19,8 ±4,3	18,1 ±5,5	17,9 ±5,3	19,3 ±5,3	9 ±2,4	8 ±2,5	8 ±1,4

(1º Subestágio) Exp. Berto B - Objeto se movimentando lentamente e de cor primária										
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10
1 - 10	7 ±4	6,4 ±3,4	8 ±4,2	8 ±3,6	6,7 ±3,5	6,9 ±3,5	5 ±3	4,1 ±2	2 ±1,4	2 ±1,5
11 - 20	19 ±7,4	19,4 ±7,4	19,9 ±7,5	19 ±7,6	18,5 ±8,8	21,9 ±9	19 ±8	7 ±3	10 ±3	6 ±2,5
21 - 30	14 ±10,4	12 ±8,2	15 ±11,5	13 ±9	15,3 ±9,9	11 ±8,5	13 ±10	9 ±6	6 ±4,9	6 ±4
31 - 40	19,8 ±8,6	21,3 ±8,5	22 ±8,6	22 ±7,6	8,8 ±20,7	18,9 ±7,4	19 ±7,5	9 ±3,4	10,9 ±4	9 ±4
41 - 50	17,3 ±6	19,4 ±6	21 ±8	19,2 ±7,5	18,5 ±7,2	17,8 ±6,5	18 ±7	9,9 ±3,9	7 ±3,5	9,9 ±3
51 - 60	5 ±2	6,5 ±2	7 ±3	7 ±2,5	4,5 ±1,8	4 ±1,5	4 ±1,5	14 ±6,5	2 ±1	3 ±1
61 - 70	15 ±6	15,4 ±6	16,9 ±6,5	17 ±6	16,2 ±8,1	16 ±7,5	3 ±1	8,9 ±3	8,9 ±5,4	6 ±2,6
71 - 80	13,7 ±5	11 ±4,3	12 ±4,6	13,2 ±5,6	11 ±5,5	12 ±5,6	10,5 ±4	6 ±2,4	5 ±2	5 ±2,5
81 - 90	20 ±7,5	15,7 ±6	17,9 ±6,5	19 ±6	19,7 ±6,3	12 ±5,6	19 ±7	8 ±3,5	9 ±4	6,9 ±2,4
91 - 100	9 ±6,6	8,4 ±5	11 ±7,5	6 ±3,7	9,8 ±6	8 ±5,8	8 ±5,9	3 ±2,5	5 ±3	3 ±1,9

Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios durante o treinamento do Aprendizado Procedimental no 2º subestágio e durante a validação com os experimentos de [Berto \(2020\)](#). a_{M1} (manter o foco); a_{M2} (*NeckYaw* p/ esquerda); a_{M3} (*NeckYaw* p/ direita); a_{M4} (*HeadPitch* p/ baixo); a_{M5} (*HeadPitch* p/ cima); a_{M6} (Fóvea p/ posição 0); a_{M7} (Fóvea p/ posição 1); a_{M8} (Fóvea p/ posição 2); a_{M9} (Fóvea p/ posição 3); a_{M10} (Fóvea p/ posição 4).

(2º Subestágio) Aprendizado Procedimental										
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10
1 - 10	14,3 ±6,9	3,1 ±3,1	17,1 ±7,3	6 ±3,7	8,1 ±3	21 ±21	12,6 ±9,8	5 ±5	6,9 ±5,4	3 ±3
11 - 20	14,9 ±9	10,7 ±6,5	12,5 ±6	7 ±2,6	10 ±4	14,3 ±10,7	11,7 ±5,7	7 ±5	4 ±2,5	6 ±5,5
21 - 30	12 ±8,4	2,9 ±1,9	8 ±5	7,1 ±4	6 ±3,5	8,2 ±7,6	8,8 ±5,2	3 ±3	3 ±3	3,2 ±2,2
31 - 40	3 ±2	2 ±2,5	2 ±1,6	4 ±2,5	4 ±2,5	3,2 ±4	3,9 ±4,5	0 ±0,5	1 ±1	0 ±0
41 - 50	15 ±7,4	7,8 ±6,1	9 ±3,2	9 ±3,1	10 ±4,6	9 ±6,5	5,8 ±4,8	4 ±2,5	5 ±6,5	4,9 ±1,9
51 - 60	6,2 ±4	4 ±3	14,2 ±7,3	6 ±3,5	6,1 ±2,6	6 ±4,6	6,7 ±6,2	5 ±3,5	5 ±4,4	1 ±1
61 - 70	17 ±7,8	7,9 ±3,5	14 ±7,2	10,1 ±2,6	7 ±3	11,2 ±6,6	9,6 ±3,8	5 ±3,5	6 ±3	3 ±1,6
71 - 80	14 ±6,4	6,9 ±2,6	5 ±2,6	6 ±2,1	10 ±3,6	12 ±5,2	8,8 ±4,4	6 ±4,5	4 ±2	3 ±2
81 - 90	22 ±8,6	8,8 ±3,4	13 ±4,7	16 ±6	14 ±6	12,2 ±2,6	9,7 ±4,8	8 ±2,5	7 ±3	5 ±2,5
91 - 100	21 ±8,4	11 ±5,5	10,2 ±5,6	10 ±4	14 ±5,6	10 ±6,4	11,6 ±4,8	7 ±4	7 ±3,4	6 ±3,5
101 - 110	13 ±6,6	6 ±3	10,2 ±4,6	9,1 ±3	11 ±4	14 ±8	8,8 ±3,4	6 ±3	4 ±2,5	6 ±4
111 - 120	29,7 ±4,7	14,8 ±8,3	16,2 ±7	18 ±8	19 ±6,6	20 ±11	19,3 ±10	7 ±4	5 ±3,9	7 ±4,4
121 - 130	21,1 ±4,5	11,9 ±4,5	11,1 ±3,1	18 ±3,5	11 ±3,7	11 ±4	12,7 ±4,4	6 ±2,5	6 ±2,5	3 ±1
131 - 140	24,8 ±15,6	11,9 ±9	18,1 ±7,2	16 ±8	12 ±9,6	13,1 ±9	10,7 ±7,2	8 ±5	8 ±4,9	7 ±5
141 - 150	19 ±9	10 ±4,5	10 ±5,7	13 ±4	12 ±6,5	16 ±6,4	13,5 ±5,4	8 ±5,5	4,9 ±3	4 ±2
151 - 160	26,4 ±8,3	11,9 ±3,9	15,2 ±5,2	14 ±4	16 ±5	18 ±6,6	12,7 ±3,4	8 ±3	6 ±2	8 ±2
161 - 170	37,8 ±18	19,8 ±5,5	17 ±8	19 ±9	21 ±8	18 ±8	20,4 ±9,7	11 ±4,5	9 ±4,5	10 ±5,5
171 - 180	25 ±10	14 ±5	12 ±4	14 ±4	13 ±6	15,2 ±5,6	16,5 ±6,2	7 ±3	6 ±3,5	7 ±2
181 - 190	25 ±10	13 ±6	11,1 ±3,6	14 ±6	18 ±8	17,1 ±8,1	14,6 ±5,9	7 ±2,5	8 ±3,1	7 ±3
191 - 200	38 ±17	19 ±8,5	18,8 ±9,8	20 ±8,1	21 ±9,5	21 ±11,5	21,3 ±10,1	11 ±5,4	8 ±3,1	8 ±2,9
(2º Subestágio) Exp. Berto - Objeto se movimentando e de cor primária										
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10
1 - 10	21,3 ±7	11,2 ±3,6	9,2 ±3	11 ±3	11 ±3,6	11 ±3,4	11,1 ±4	6 ±2	4 ±1,5	5 ±1,5
11 - 20	29,2 ±12	13,2 ±6	13,2 ±5,1	13,8 ±6	13 ±6	18 ±7	15,1 ±6	6 ±3	5 ±2,6	7 ±4
21 - 30	27,3 ±11,7	13,2 ±5,6	16,4 ±7,7	15,8 ±8	17,9 ±7,5	19,8 ±9,4	18,2 ±9	7 ±3,5	8,1 ±3	8 ±3,4
31 - 40	21,4 ±11	11,2 ±6,2	12,2 ±6	10,8 ±5,4	10,9 ±5	12 ±7	12,1 ±7	4 ±2	5 ±3	4 ±3
41 - 50	25,3 ±7,8	13,2 ±4,5	13,3 ±3,5	13,8 ±3,9	10,9 ±4	13 ±4	14 ±4,5	4 ±1,5	6 ±1,6	7 ±2,5
51 - 60	35,4 ±10	21,2 ±6,6	18,4 ±5,6	14,9 ±5	16,8 ±5	17,9 ±6,4	16 ±5	8 ±3	6 ±1,6	9 ±3
61 - 70	27,3 ±9,2	13,2 ±4,6	16,2 ±6,6	15 ±5	11 ±5	15,9 ±5	10,2 ±4,5	5 ±2	7,2 ±2	5 ±1,9
71 - 80	27,3 ±13,5	13,2 ±7,2	13,2 ±6	10,9 ±5,5	13 ±6,6	15 ±7,5	13 ±7	6 ±3,4	7,1 ±4	6 ±3,5
81 - 90	22,3 ±10	11 ±6,6	10,2 ±4,1	10 ±3,6	10 ±4,5	12,9 ±5	9 ±4	4 ±1,5	5 ±1,5	5 ±2
91 - 100	28,3 ±4,8	15,3 ±4,6	17,3 ±3	16,8 ±3	16,9 ±3	16,9 ±3,4	15 ±5	7 ±2	7 ±2	8 ±1,5

Médias e desvios padrão do número de ações executadas para cada possível ação a cada 10 episódios durante o treinamento do Aprendizado Procedimental no 3º subestágio e durante a validação com os experimentos de [Berto \(2020\)](#). a_{M1} (manter o foco); a_{M2} (*NeckYaw* p/ esquerda); a_{M3} (*NeckYaw* p/ direita); a_{M4} (*HeadPitch* p/ baixo); a_{M5} (*HeadPitch* p/ cima); a_{M6} (Fóvea p/ posição 0); a_{M7} (Fóvea p/ posição 1); a_{M8} (Fóvea p/ posição 2); a_{M9} (Fóvea p/ posição 3); a_{M10} (Fóvea p/ posição 4); a_{M11} (*NeckYaw* p/ foco atencional); a_{M12} (*NeckYaw* p/ longe do foco atencional); a_{M13} (*HeadPitch* p/ foco atencional); a_{M14} (*HeadPitch* p/ longe do foco atencional); a_{A1} (mover foco atencional de acordo com $\mathcal{F}_5$); a_{A2} (mover foco atencional de acordo com $\mathcal{F}_6$); a_{A3} (mover foco atencional de acordo com $\mathcal{F}_7$).

(3º Subestágio) Aprendizado Procedimental																	
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10	AM11	AM12	AM13	AM14	AA1	AA2	AA3
1 - 10	6 ± 6	2 ± 1	5 ± 4,6	9 ± 4,6	5,9 ± 4	4,9 ± 3	13,9 ± 9	8,9 ± 6,5	22 ± 8,4	3 ± 3	3,9 ± 2,6	2 ± 1,5	2 ± 1,5	4,2 ± 2,7	4,9 ± 4	3 ± 2,6	13,2 ± 13,2
11 - 20	5 ± 5	1 ± 1	3 ± 3	1,1 ± 1	1 ± 1	2 ± 2	3 ± 3	1 ± 0,5	9 ± 6,5	3 ± 3	2 ± 3	0 ± 0	6 ± 3,5	8 ± 3	2 ± 3	3 ± 3,5	2 ± 3,3
21 - 30	5 ± 2,5	7 ± 5,5	6 ± 3,5	5,9 ± 4	8,8 ± 5,2	5,1 ± 3	9,9 ± 6,9	2 ± 1,5	3 ± 3	3,2 ± 2,2	6 ± 2	4 ± 3,4	4 ± 3	8 ± 5	5 ± 5	2 ± 1,5	8 ± 5,6
31 - 40	2 ± 1	2 ± 1,6	2 ± 1,5	5 ± 2,5	4,9 ± 2,5	7,8 ± 4	2 ± 1,4	3 ± 1,5	3 ± 2	0 ± 0	5,9 ± 2	4 ± 2	6 ± 3	4 ± 2,5	7 ± 4	2 ± 2	7,3 ± 5,7
41 - 50	5 ± 3	5,9 ± 3,5	2 ± 1	6 ± 4	4 ± 1,5	8,8 ± 6,4	5 ± 4	9 ± 6,5	4 ± 2,3	4,9 ± 1,9	5 ± 2,6	8 ± 3	6 ± 2	3,1 ± 1,6	5,1 ± 4,1	5 ± 3	4 ± 3
51 - 60	8 ± 5	3 ± 2	5 ± 4	9,1 ± 4,6	8,9 ± 5,5	8,8 ± 5,9	7 ± 4,5	6 ± 5,4	7 ± 6,1	6 ± 6	7 ± 4	6 ± 4,5	9 ± 7	10 ± 5,6	8 ± 4,7	6 ± 5	6 ± 5,6
61 - 70	9 ± 6,4	5 ± 4,4	7 ± 5	7 ± 5,5	5 ± 3,5	5,9 ± 4,3	6,9 ± 4,9	8 ± 5,4	11 ± 8,5	9 ± 6,3	6 ± 6,9	10 ± 7,4	7 ± 3,5	7 ± 6	3,1 ± 3	6 ± 4	7 ± 5,7
71 - 80	5 ± 2,5	5 ± 2,6	5 ± 3,5	5,1 ± 2	10,8 ± 4,4	7,9 ± 3,4	5,9 ± 4,5	9 ± 4	10 ± 4,5	4 ± 2	8 ± 3,6	9 ± 2,6	7 ± 4,5	9 ± 3,5	10 ± 4	7 ± 4	8 ± 3,6
81 - 90	6 ± 2,5	8 ± 5,5	5 ± 2	9 ± 4	5 ± 2,6	6 ± 2,5	5,9 ± 4,5	7 ± 3,5	5 ± 3	5 ± 2,4	4 ± 2	5 ± 4	3 ± 1,5	8 ± 2,5	8 ± 5	4 ± 2	3 ± 2
91 - 100	9 ± 6	9 ± 4	14 ± 5	14 ± 6,5	11,9 ± 5,9	12,7 ± 2,9	11 ± 5,3	15,9 ± 7,1	13 ± 5,5	16 ± 6,5	9 ± 4	16 ± 6,1	17 ± 8	12 ± 6	10,1 ± 5,7	11 ± 4,6	9 ± 4
101 - 110	12 ± 4,4	7 ± 2,5	11 ± 5	11 ± 4	14,9 ± 4,8	12,9 ± 4,4	6,9 ± 3	10 ± 3,8	9 ± 2	11 ± 4	11 ± 4	11 ± 4,6	14 ± 4,5	14 ± 4,5	13 ± 5,2	8 ± 4	13 ± 6,6
111 - 120	13 ± 7	7 ± 4	9 ± 5	13,2 ± 4,6	10 ± 5	7 ± 4	11 ± 6,5	8,1 ± 5,5	6,1 ± 2,1	9 ± 5,5	7 ± 3,5	8 ± 5	8 ± 4	8 ± 4	7,2 ± 4,7	11 ± 5,7	9 ± 5
121 - 130	13 ± 4	11 ± 3,6	12 ± 4	12 ± 4,5	13,9 ± 4,5	6 ± 2,5	11,9 ± 3,9	12 ± 4,4	12 ± 3,4	10 ± 5,5	14,9 ± 5,3	15 ± 4,2	11 ± 3	11 ± 3	11 ± 3,7	8,5 ± 4,3	12 ± 3,5
131 - 140	9 ± 2,3	7 ± 2,5	7 ± 2	11 ± 2,6	9 ± 3,5	8,9 ± 4,9	11,9 ± 4,4	8 ± 2,5	8 ± 3	10 ± 2	8 ± 4	10 ± 3	11 ± 4,4	9 ± 2,6	10,1 ± 3,5	11 ± 3,5	9 ± 3
141 - 150	10 ± 3	7 ± 2	8 ± 3,5	12,1 ± 2,5	10 ± 2,5	7,8 ± 3	7,9 ± 2,9	10 ± 3	8,9 ± 2,5	10 ± 3,5	8 ± 3,5	8 ± 2	11 ± 2	6 ± 1,6	9 ± 2,5	6 ± 2	10 ± 4,6
151 - 160	9 ± 3,5	9 ± 2,9	9 ± 4	11,1 ± 3,5	11 ± 4,5	8,9 ± 3,5	7,9 ± 3,5	8 ± 2,5	8,9 ± 2,5	10 ± 3,5	9 ± 3,5	7 ± 3	12 ± 4,5	9 ± 3,5	9 ± 4,5	7 ± 3	9 ± 3,6
161 - 170	16 ± 4,5	13,9 ± 4,5	17 ± 6,6	11 ± 4,5	14 ± 5	15,8 ± 5,4	11,8 ± 3,4	16 ± 5,6	14,9 ± 5,5	14 ± 4,5	15 ± 5,1	13 ± 4,6	18 ± 6	13 ± 4,6	13 ± 5	13 ± 5	13 ± 4,7
171 - 180	9 ± 5	9 ± 4	7 ± 3	11 ± 6	9 ± 4	7,9 ± 4	11 ± 5	9,1 ± 4,9	14,9 ± 5,5	13 ± 7	8 ± 4,5	9 ± 4,6	10 ± 5,5	9 ± 5,5	9 ± 4,6	10 ± 5,5	8 ± 4,6
181 - 190	10 ± 1,4	9 ± 1,6	11 ± 2	11 ± 2,5	13 ± 2	8 ± 5,6	12 ± 2,5	12 ± 1,4	11 ± 2	10 ± 1,5	11 ± 2,1	10 ± 1,9	10 ± 1,7	9 ± 1,7	10 ± 1,3	10 ± 1	12 ± 2,3
191 - 200	18,9 ± 4,5	17 ± 5	16 ± 5	19 ± 5,6	16,9 ± 5,1	15,7 ± 6	17,8 ± 5	18 ± 5,4	20 ± 5,5	17,9 ± 5,5	16,9 ± 5	15 ± 5	18,9 ± 5,5	20 ± 5	17 ± 4,5	16 ± 4	16 ± 5
(3º Subestágio) Exp. Berto - Objeto se movimentando e de cor primária																	
Episódios	AM1	AM2	AM3	AM4	AM5	AM6	AM7	AM8	AM9	AM10	AM11	AM12	AM13	AM14	AA1	AA2	AA3
1 - 10	14,1 ± 8	9,8 ± 6,4	13 ± 7	10 ± 5	12 ± 7	13,2 ± 6,5	11 ± 5,9	9 ± 5,1	8 ± 5	12 ± 7,6	8,1 ± 5,6	9 ± 5,5	10 ± 5,9	10 ± 6	8,1 ± 5,2	8 ± 4,6	10 ± 6
11 - 20	18 ± 7	14,9 ± 5	15 ± 5	15 ± 5	14 ± 5,1	17 ± 7,6	15 ± 5,5	13 ± 4,5	13 ± 5,5	16 ± 6,5	15 ± 5,1	15 ± 5,6	14 ± 4,6	19,9 ± 4,6	13,1 ± 4,5	13 ± 5	12,9 ± 6
21 - 30	10 ± 4	8 ± 4,6	12 ± 6	7 ± 2,6	9 ± 4,1	10 ± 4	9,1 ± 3,5	6 ± 3	7 ± 3,6	9 ± 4,4	8 ± 3,6	9 ± 3,6	7 ± 2,6	6,2 ± 3	8 ± 4	7,1 ± 3,6	8 ± 3,1
31 - 40	14 ± 4,5	11 ± 4	12 ± 3,6	12 ± 4	14 ± 4	14 ± 5,6	12,1 ± 4,5	10,1 ± 4	12 ± 4,6	14 ± 4,6	9 ± 4	11 ± 4	12 ± 4,5	9 ± 3,5	8 ± 3	11 ± 3,6	9 ± 3
41 - 50	16,1 ± 6	15 ± 5,6	15 ± 5	15 ± 4,5	14 ± 5,6	15 ± 5	15 ± 5,1	12,1 ± 4,6	13 ± 4,5	17 ± 6	13 ± 3,5	11 ± 4	14,1 ± 5,6	10,9 ± 2,9	11,1 ± 4	10 ± 3,6	12 ± 4,5
51 - 60	18 ± 4,5	15,8 ± 4	21 ± 6	17 ± 4	18 ± 5	16,9 ± 4	18 ± 5	16,1 ± 5	15 ± 4,5	21 ± 5	16 ± 4	15,1 ± 3,6	17 ± 4,5	15 ± 5,1	14 ± 3,5	17 ± 5	16 ± 5
61 - 70	16 ± 5	16 ± 5,5	17 ± 5	16 ± 5,6	16 ± 5,6	18,9 ± 7	14,1 ± 6,5	13,1 ± 4,5	17 ± 7	18,2 ± 6	13 ± 4	13 ± 5	16 ± 5,5	14 ± 5	14,1 ± 5	12 ± 5	14 ± 6,5
71 - 80	14 ± 3,5	9,9 ± 2,5	15 ± 4,5	13 ± 3,5	15 ± 4,7	16 ± 4,5	11 ± 3,5	12,1 ± 3	12 ± 4	15 ± 3,5	10,9 ± 3	13 ± 4,1	13,9 ± 4	9 ± 2,6	12 ± 3,5	13 ± 3	12 ± 3,5
81 - 90	15 ± 5	11,9 ± 4,5	14 ± 5	12 ± 4	14 ± 4	10 ± 2,8	11 ± 3	11,1 ± 4	11 ± 3	15 ± 4,5	7,9 ± 3	10 ± 2,5	12 ± 3	10 ± 3	9 ± 2	10 ± 2,5	12 ± 3,6
91 - 100	7 ± 5,3	6 ± 4,8	9 ± 7,6	6 ± 4,6	8 ± 5,7	7 ± 4,9	8,1 ± 6	7 ± 6	5 ± 3,6	6 ± 5	6 ± 4,9	6 ± 5	8 ± 5,5	6 ± 5	8 ± 7,2	5,2 ± 3,7	6,9 ± 5,6