List of Equations in
Causal Cognitive Architecture 3: A Solution to the Binding Problem
Cognitive Systems Research, in press

Howard Schneider
Sheppard Clinic North

The variables and parameters below are discussed in the above referenced article. This list is simply intended as a supplementary resource while reading the paper. For reasons of brevity, dot notation is used in algorithmic portions of the equations—this is discussed in the paper. As well, the full algorithmic expansion of the dot notation is provided by the supplementary Python code.

self.all_maps = np.empty((6, 6, 6, 30 * SECONDS_TO_RUN, 1000000, 50),dtype=object) (1)

self.all_maps = np.empty((6, 6, 6, 1000000, 50),dtype=object) (2)

$$\mathbf{S_1} \in R^{m\_1 x n\_1 x o\_1} \quad (3)$$

$$\mathbf{S_{1,t}} := \text{visual inputs(t)} \quad (4)$$

$$\mathbf{S_2} \in R^{m\_2 x n\_2 x o\_2} \quad (5)$$

$$\mathbf{S_{2,t}} := \text{auditory inputs(t)} \quad (6)$$

$$\mathbf{S_3} \in R^{m\_3 x n\_3 x o\_3} \quad (7)$$

$$\mathbf{S_{3,t}} := \text{olfactory inputs(t)} \quad (8)$$

$$\boldsymbol{\sigma} := \text{sensory system identification code} \in N \quad (9)$$

$$\boldsymbol{n\_\sigma} := \text{total number of sensory systems} \in N \quad (10)$$

$$\boldsymbol{s(t)} = [\, \mathbf{S_{1,t}}, \mathbf{S_{2,t}}, \mathbf{S_{3,t}}, \dots, \mathbf{S_{n\_\sigma,t}}] \quad (11)$$

$$\boldsymbol{s'(t)} = \text{Input\_Sensory\_Vectors\_Shaping\_Modules}(\, \boldsymbol{s(t)} \,) = [\mathbf{S'_{1,t}}, \mathbf{S'_{2,t}}, \mathbf{S'_{3,t}}, \dots, \mathbf{S'_{n\_\sigma,t}}] \quad (12)$$

$$\mathbf{S'_{\sigma,t}} \in R^{mxnxo} \quad (13)$$

$$\mathbf{mapno} := \text{map identification code} \in N \quad (14)$$

$$\boldsymbol{\Theta} := \text{total number of local navigation maps in a sensory system } \boldsymbol{\sigma} \in N \quad (15)$$

$$\mathbf{LNM_{(\sigma,mapno)}} \in R^{mxnxo} \quad (16)$$

$$\boldsymbol{all\_maps_{\sigma,t}} = [\mathbf{LNM_{(\sigma,1,t)}}, \mathbf{LNM_{(\sigma,2,t)}}, \mathbf{LNM_{(\sigma,3,t)}}, \dots, \mathbf{LNM_{(\sigma,\Theta,t)}}] \quad (17)$$

$$\Upsilon := \textbf{mapno} \text{ of best matching map in a given set of navigation maps} \in \textbf{mapno} \quad (18)$$

$$\textbf{LNM}_{(\sigma,\Upsilon,t)} = \text{Input\_Sensory\_Vectors\_Associations\_Module}_{\sigma}.\text{match\_best\_local\_navigation\_map}(\,\textbf{S'}_{\sigma,t}\,) \quad (19)$$

$$\textbf{h} = \text{number of differences allowed to be copied onto existing map} \in R \quad (20a)$$

$$\textbf{new\_map} := \textbf{mapno} \text{ of new local navigation map added to current sensory system } \boldsymbol{\sigma} \in \textbf{mapno} \quad (20b)$$

$$|\text{ differences } (\textbf{S'}_{\sigma,t}, \textbf{LNM}_{(\sigma,\Upsilon,t)})\,| \le \textbf{h}, \Rightarrow \textbf{LNM'}_{(\sigma,\Upsilon,t)} = \textbf{LNM}_{(\sigma,\Upsilon,t)} \cup \textbf{S'}_{\sigma,t} \quad (21)$$

$$|\text{ differences } (\textbf{S'}_{\sigma,t}, \textbf{LNM}_{(\sigma,\Upsilon,t)})\,| > \textbf{h}, \Rightarrow \textbf{LNM'}_{(\sigma,\Upsilon,t)} = \textbf{LNM}_{(\sigma,\text{new\_map},t)} \cup \textbf{S'}_{\sigma,t} \quad (22)$$

$$\boldsymbol{lnm_t} = [\textbf{LNM'}_{(1,\Upsilon,t)}, \textbf{LNM'}_{(2,\Upsilon,t)}, \textbf{LNM'}_{(3,\Upsilon,t)}, \ldots, \textbf{LNM'}_{(n\_\sigma,\Upsilon,t)}] \quad (23)$$

$$\textbf{NM}_{\textbf{mapno}} \in R^{mxnxo}, \textbf{IPM}_{\textbf{mapno}} \in R^{mxnxo}, \textbf{LPM}_{\textbf{mapno}} \in R^{mxnxo} \quad (24)$$

$$\boldsymbol{\Theta}\textbf{\_NM} := \text{total NM's} \in N, \ \boldsymbol{\Theta}\textbf{\_IPM} := \text{total IPM's} \in N, \ \boldsymbol{\Theta}\textbf{\_LPM} := \text{total LPM's} \in N \quad (25)$$

$$\boldsymbol{all\_LNMs_t} := [\boldsymbol{all\_maps_{1,t}}, \boldsymbol{all\_maps_{2,t}}, \boldsymbol{all\_maps_{3,t}}, \ldots, \boldsymbol{all\_maps_{n\_\sigma,t}}] \quad (26)$$

$$\boldsymbol{all\_NMs_t} := [\textbf{NM}_{1,t}, \textbf{NM}_{2,t}, \textbf{NM}_{3,t}, \ldots, \textbf{NM}_{\Theta\_NM,t}] \quad (27)$$

$$\boldsymbol{all\_IPMs_t} := [\textbf{IPM}_{1,t}, \textbf{IPM}_{2,t}, \textbf{IPM}_{3,t}, \ldots, \textbf{IPM}_{\Theta\_IPM,t}] \quad (28)$$

$$\boldsymbol{all\_LPMs_t} := [\textbf{LPM}_{1,t}, \textbf{LPM}_{2,t}, \textbf{LPM}_{3,t}, \ldots, \textbf{LPM}_{\Theta\_LPM,t}] \quad (29)$$

$$\boldsymbol{all\_navmaps_t} := [\boldsymbol{all\_LNMs_t}, \boldsymbol{all\_NMs_t}, \boldsymbol{all\_IPMs_t}, \boldsymbol{all\_LPMs_t}] \quad (30)$$

$$\boldsymbol{modcode} := \text{module identification code} \in N \quad (31)$$

$$\boldsymbol{mapcode} := [\boldsymbol{modcode}, \boldsymbol{mapno}] \quad (32)$$

$$\boldsymbol{\chi} := [\boldsymbol{mapcode}, \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}] \quad (33)$$

$$\boldsymbol{feature} \in R, \ \boldsymbol{action} \in R \quad (34)$$

$\boldsymbol{\Phi}$**_feature** := last **feature** contained by a cube, $\boldsymbol{\Phi}$**_action** := last **action** contained by a cube, $\boldsymbol{\Phi\_\chi}$:= last χ (i.e., address) contained by a cube (35)

$$\boldsymbol{cubefeatures_{\chi,t} :=} [\boldsymbol{feature_{1,t}},\ \boldsymbol{feature_{2,t}},\ \boldsymbol{feature_{3,t}},\ \ldots,\ \boldsymbol{feature_{\Phi\_feature,t}}] \quad (36)$$

$$\boldsymbol{cubeactions_{\chi,t} :=} [\boldsymbol{action_{1,t}},\ \boldsymbol{action_{2,t}},\ \boldsymbol{action_{3,t}},\ \ldots,\ \boldsymbol{action_{\Phi\_action,t}}] \quad (37)$$

$$\boldsymbol{linkaddresses_{\chi,t} :=} [\boldsymbol{\chi_{1,t}}, \boldsymbol{\chi_{2,t}}, \boldsymbol{\chi_{3,t}},\ \ldots,\ \boldsymbol{\chi_{\Phi\_\chi,t}}] \quad (38)$$

$$\boldsymbol{cubevalues_{\chi,t} :=}\ [\boldsymbol{cubefeatures_{\chi,t}}, \boldsymbol{cubeactions_{\chi,t}}, \boldsymbol{linkaddresses_{\chi,t}}] \quad (39)$$

$$\boldsymbol{cubevalues_{\chi,t}} = \boldsymbol{all\_navmaps_{\chi,t}} \ (40)$$

$$\boldsymbol{linkaddresses_{\chi,t}} = \boldsymbol{link}(\boldsymbol{\chi,t}) \ (41)$$

$$\boldsymbol{grounded\_feature}\ :=\ \forall_{\boldsymbol{feature}}:\ \boldsymbol{feature} \in \boldsymbol{all\_LNMs_{\chi}} \quad (42)$$

$$\forall_{\chi,t}: \boldsymbol{all\_navmaps_{\chi,t} = grounded\_feature} \text{ OR } \boldsymbol{link(all\_navmaps}_{\chi,t}) \neq [\ ] \text{ OR } \boldsymbol{all\_navmaps_{\chi,t}} = [\ ] \quad (43)$$

$$\boldsymbol{s'\_series(t)} = [\boldsymbol{s'(t\text{-}3)}, \boldsymbol{s'(t\text{-}2)}, \boldsymbol{s'(t\text{-}1)}, \boldsymbol{s'(t)}] \quad (44)$$

visual_series(t) = Sequential/Error_Correcting_Module**.**visual_inputs(***s'_series(t)***) (45)

auditory_series(t) = Sequential/Error_Correcting_Module**.**auditory_inputs(***s'_series(t)***) (46)

visual_motion(t) = Sequential/Error_Correcting_Module**.**visual_match(***visual_series(t)***) (47)

auditory_motion(t) = Sequential/Error_Correcting_Module**.**auditory_match(***auditory_series(t)***) (48)

$$\mathbf{VNM} \in \mathrm{R}^{mxnxo},\ \ \mathbf{AVNM} \in \mathrm{R}^{mxnxo} \quad (49)$$

$$\mathbf{VNM'_t} = \mathbf{VNM_t} \cup \boldsymbol{visual\_motion(t)} \quad (50a)$$

$$\mathbf{VNM''_t} = \mathbf{VNM'_t} \cup \boldsymbol{auditory\_motion(t)} \quad (50b)$$

$\mathbf{AVNM_t}$ = Sequential/Error_Correcting_Module**.**auditory_match_process(***auditory_series(t)***) (51)

$$\mathbf{VSNM} \in \mathrm{R}^{mxnxo} \quad (52)$$

visual_segmented_series(t) = [$\mathbf{VSNM_{t-3}}$, $\mathbf{VSNM_{t-2}}$,$\mathbf{VSNM_{t-1}}$, and $\mathbf{VSNM_t}$] (53)

visseg_motion(t) = Sequential/Error_Correcting_Module**.**visual_match(***visual_segmented_series(t)***) (54)

$$\mathbf{VSNM'_t = VSNM_t} \cup \boldsymbol{visseg\_motion(t)} \quad (55)$$

$$\mathbf{LNM'_{(1, \Upsilon, t)}} = \boldsymbol{lnm_t}[0] \quad (56)$$

$$\mathbf{CONTEXT} := \in \mathrm{R}^{mxnxo} \quad (57)$$

$$\mathbf{WNM} := \in \mathrm{R}^{mxnxo} \quad (58)$$

$$\mathbf{CONTEXT_t = WNM_t} \quad (59)$$

$$\mathbf{VSNM_t} = \text{Object\_Segmentation\_Gateway\_Module.visualsegment}(\mathbf{LNM'_{(1, \Upsilon, t)}}, \mathbf{VNM''_t}, \mathbf{CONTEXT_t}) \quad (60)$$

$$\mathbf{WNM_t} = \text{Causal\_Memory\_Module.match\_best\_multisensory\_navigation\_map}(\mathbf{VSNM'_t}, \mathbf{AVNM_t}, \mathbf{LNM'_{(3, \Upsilon, t)}}, \mathbf{LNM'_{(4, \Upsilon, t)}}, \ldots, \mathbf{LNM'_{(\boldsymbol{n\_\sigma}, \Upsilon, t)}}) \quad (61)$$

$$\mathbf{h'} = \text{number of differences allowed to be copied onto existing navigation map} \in \mathrm{R} \quad (62)$$

$$\boldsymbol{actual_t} = [\mathbf{VSNM'_t}, \mathbf{AVNM_t}, \mathbf{LNM'_{(3, \Upsilon, t)}}, \mathbf{LNM'_{(4, \Upsilon, t)}}, \ldots, \mathbf{LNM'_{(\boldsymbol{n\_\sigma}, \Upsilon, t)}}] \quad (63)$$

$$\mathbf{NewNM} \in \mathrm{R}^{mxnxo} \quad (64)$$

$$|\,\mathbf{differences}(\boldsymbol{actual_t}, \mathbf{WNM_t})\,| \leq \mathbf{h'}, \Rightarrow \mathbf{WNM'_t = WNM_t} \cup \boldsymbol{actual_t} \quad (65)$$

$$|\,\mathbf{differences}(\boldsymbol{actual_t}, \mathbf{WNM_t})\,| > \mathbf{h'}, \Rightarrow \mathbf{WNM'_t = NewNM_t} \cup \boldsymbol{actual_t} \quad (66)$$

$$\boldsymbol{emotion} \in \mathrm{R} \quad (67)$$

$$\mathbf{GOAL} \in \mathrm{R}^{mxnxo} \quad (68)$$

$$\boldsymbol{autonomic} \in \mathrm{R} \quad (69)$$

$$[\boldsymbol{emotion_t}, \mathbf{GOAL_t}] = \text{Goal/Emotion\_Module.set\_emotion\_goal}(\boldsymbol{autonomic_t}, \mathbf{WNM'_t}) \quad (70)$$

$$\mathbf{WIP} \in \mathrm{R}^{mxnxo} \quad (71)$$

$$\mathbf{WIP_t} = \text{Instinctive\_Primitives\_Module.match\_best\_primitive}(\boldsymbol{actual_t}, \boldsymbol{emotion_t}, \mathbf{GOAL_t}) \quad (72)$$

$$\mathbf{WLP} \in \mathrm{R}^{mxnxo} \quad (73)$$

$$\mathbf{WLP_t} = \text{Learned\_Primitives\_Module.match\_best\_primitive}(\boldsymbol{actual_t}, \boldsymbol{emotion_t}, \mathbf{GOAL_t}) \quad (74)$$

$$\mathbf{WPR} \in \mathrm{R}^{mxnxo} \quad (75)$$

$$\mathbf{WLP_t} = [\ ], \Rightarrow \mathbf{WPR_t = WIP_t} \quad (76)$$

$$\mathbf{WLP_t} \neq [\ ],\ \Rightarrow \mathbf{WPR_t} = \mathbf{WLP_t} \quad (77)$$

$$\boldsymbol{action_t} = \text{Navigation\_Module.apply\_primitive}(\mathbf{WPR_t}, \mathbf{WNM'_t}) \quad (78)$$

$$\boldsymbol{output\_vector} \in \mathrm{R}^{n'} \quad (79)$$

$$\boldsymbol{action_t} = [\text{"move*"}],\ \Rightarrow \boldsymbol{output\_vector_t} = \text{Output\_Vector\_Association\_Module.}$$
$$\text{action\_to\_output}(\ \boldsymbol{action_t}, \mathbf{WNM'_t}\ ) \quad (80)$$

$$\boldsymbol{motion\_correction} \in \mathrm{R}^2 \quad (81)$$

$$\mathbf{action_t} = [\text{"move*"}],\ \Rightarrow \boldsymbol{motion\_correction_t} = \text{Sequential/Error\_Correcting\_Module.}$$
$$\text{motion\_correction}(\ \boldsymbol{action_t}, \mathbf{WNM'_t}, \boldsymbol{visual\_series(t)}\ ) \quad (82)$$
$$\boldsymbol{output\_vector'_t} = \text{Output\_Vector\_Association\_Module.}$$
$$\text{apply\_motion\_correction}(\ \boldsymbol{output\_vector_t}, \boldsymbol{motion\_correction_t}\ ) \quad (83)$$

$$\boldsymbol{explanation} \in \mathrm{R}^{n''} \quad (84)$$

$$\boldsymbol{explanation_t} = \text{Navigation\_Module.navmap\_to\_proto\_lang}(\ \mathbf{WPR_t}, \mathbf{WNM'_t},\ \boldsymbol{action_t}\ ) \quad (85)$$

$$\mathbf{LNM_{(\sigma, \Upsilon, t)}} = \text{Input\_Sensory\_Vectors\_Associations\_Module}_{\boldsymbol{\sigma}}\text{.}$$
$$\text{match\_best\_local\_navigation\_map}(\ \mathbf{S'_{\sigma,t}}, \mathbf{WNM'_{t-1}}\ ) \quad (86)$$

$$\mathbf{WPR_{t-1}} = [\text{"feedback intermediate*"}],\ \Rightarrow \forall_{\boldsymbol{\sigma}} : \mathbf{LNM_{(\sigma, \Upsilon, t)}} =$$
$$\text{Input\_Sensory\_Vectors\_Associations\_Module}_{\boldsymbol{\sigma}}\text{.extract\_}\sigma(\ \mathbf{WNM'_{t-1}}\ ) \quad (87)$$

$$\boldsymbol{wnm\_stack} \in \mathrm{R}^{n'''} \quad (88)$$

$$\mathbf{WNM'_t} = \text{Navigation\_Module.push\_stack}(\ \boldsymbol{wnm\_stack_t}\ ) \quad (89)$$

$$\mathbf{WNM'_t} = \text{Navigation\_Module.pop\_stack}(\ \boldsymbol{wnm\_stack_t}\ ) \quad (90)$$

$$\boldsymbol{link(\chi,t)} \in \boldsymbol{all\_LNMs_t}\ \ \mathbf{OR}\ \boldsymbol{link(\chi,t)} \in \boldsymbol{all\_NMs_t},$$
$$\Rightarrow \mathbf{WNM'_t} = \text{Navigation\_Module.retrieve\_map}(\ \boldsymbol{link(\chi,t)}\ ) \quad (91)$$

$$\boldsymbol{link(\chi,t)} \in \boldsymbol{all\_IPMs_t}\ \ \mathbf{OR}\ \boldsymbol{link(\chi,t)} \in \boldsymbol{all\_LPMs_t},$$
$$\Rightarrow \mathbf{WPR_t} = \text{Navigation\_Module.retrieve\_map}(\ \boldsymbol{link(\chi,t)}\ ) \quad (92)$$

$$\mathbf{\Gamma_{cca3}} = \boldsymbol{\eta}\boldsymbol{\rho} + \sum \boldsymbol{\rho}' \quad \text{(i)}$$

$$\mathbf{e_{cca3}} = 1/\ \mathbf{\Gamma_{cca3}} \quad \text{(ii)}$$

$$\mathbf{\Gamma_{cca1}} = \boldsymbol{\tau}\boldsymbol{\mu} + \sum \boldsymbol{\mu}' \quad \text{(iii)}$$

$$\mathbf{e_{cca1}} = 1/\ \mathbf{\Gamma_{cca1}} \quad \text{(iv)}$$

$$\mathbf{\Gamma_{cca3}} = 60\boldsymbol{\rho} \quad \text{(v)}$$

cca1_sensory_vector, ***cca1_other_vectors***, ***all_cca1_sensory_vectors***, ***cca1_binding_vector***, ***cca1_working_primitive***, ***cca1_action*** $\in R^{n''''}$ (93)

cca1_sensory_vector$_{\sigma,t}$ **=** Input_Sensory_Vectors_Associations_Module**.** match_with_other_vectors ($\mathbf{S'}_{\sigma,t}$, ***cca1_other_vectors***$_t$) (94)

all_cca1_sensory_vectors$_t$:= [***cca1_sensory_vector***$_{1,t}$, ***cca1_sensory_vector***$_{2,t}$, **…**, ***cca1_sensory_vector***$_{n\_\sigma,t}$] (95)

cca1_binding_vector$_t$ **=** Sensory_Vectors_Binding_Module**.** match_with_other_vectors (***all_cca1_sensory_vectors***$_t$, ***cca1_other_vectors***$_t$) (96)

cca1_working_primitive$_t$ = (Instinctive_Primitives_Module.match_primitive(***cca1_binding_vector***$_t$) OR Instinctive_Primitives_Module.match_primitive(***cca1_binding_vector***$_t$)) (97)

cca1_ action$_t$:= Navigation_Module**.**all_steps_to_produce_action (***cca1_binding_vector***$_t$, ***cca1_working_primitive***$_t$) (98)

$$\mathbf{\Gamma_{cca1}} = 5\boldsymbol{\mu} + 2\lambda! \quad \text{(vi)}$$

$$\mathbf{\Gamma_{cca1}} > \mathbf{\Gamma_{cca3}} \quad \text{(vii)}$$

$$\mathbf{e_{cca3}} > \mathbf{e_{cca1}} \quad \text{(viii)} \quad \square$$