Autonomous DOM Neural Unit: Core Mathematical Workflow

December 21, 2025

Contents

1 Introduction

This document presents the **exact mathematical workflow** of the DOM neural unit. The system processes DOM elements through a 6-phase cycle using matrix operations, eigen decompositions, and relational transformations.

2 Core Mathematical Framework

2.1 Dimensional Spaces

- **25D Attribute Space**: $\mathbb{A} = \mathbb{R}^{25}$
- **87D Relational Space**: $\mathbb{R}^{87}$ (binary encoding)
- **5 Patterns**: DATA_INPUT, ACTION_ELEMENT, CONTEXT_ELEMENT, STRUCTURAL, UNKNOWN
- **6 Positions**: self, parent, up, down, left, right

2.2 Key Matrices and Tensors

Symbol	**Dimensions**	**Description**
X	5×25	Self expectations for all patterns
P_i	5×25	Neighbor expectations for pattern i
B	5×5	Position bias matrix
b	5×1	Pattern probability vector
$T(V)$	$n \times 87$	T-transform of n vectors
E	$5 \times 6 \times 25$	Complete expectation tensor
T_{exp}	$5 \times 6 \times 87$	T-transformed expectations

Table 1: Core mathematical objects

3 The T-Transformation: Relational Encoding

3.1 Definition

$$T : \mathbb{R}^{n\times 25} \to \mathbb{R}^{n\times 87}$$

3.2 Composition

$$87 = \underbrace{15}_{\text{combinations}} + \underbrace{9 \times 8}_{\text{base} \times \text{questions}}$$

3.3 Example: 3 Vectors Transformation

Given $V \in \mathbb{R}^{3\times 25}$:

$$V = \begin{bmatrix} v_{1,0} & v_{1,1} & \cdots & v_{1,24} \\ v_{2,0} & v_{2,1} & \cdots & v_{2,24} \\ v_{3,0} & v_{3,1} & \cdots & v_{3,24} \end{bmatrix}$$

For dimension $d = 0$:

1. Count vectors with value $v_{1,0}$: e.g., 2 vectors
2. Set sharing flags: $\mathrm{Q3}_d = 1$ for those 2 vectors
3. Set uniqueness flag: $\mathrm{Q4}_d = 1$ for remaining vector
4. If $v_{1,0} = 0.5$, count optionality: e.g., 1 vector
5. Set optionality flag: $\mathrm{Q8}_d = 1$ for that vector

4 Complete 6-Phase Mathematical Workflow

4.1 Workflow Diagram

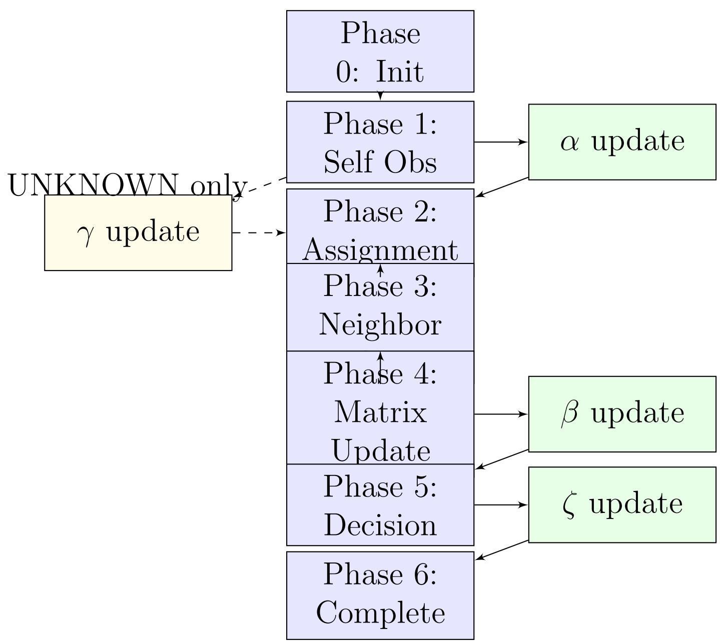

Figure 1: 6-phase workflow with eigen updates

Operation	Mathematical Result
Create ROSE instance	$\mathrm{ROSE}(p_0, c)$
Extract E tensor	$E \in \mathbb{R}^{5\times6\times25}$
Pre-compute T_{exp}	$T_{\text{exp}} \in \mathbb{R}^{5\times6\times87}$
Initialize b	$b_{\text{initial}} = [0.2, 0.2, 0.2, 0.2, 0.2]^T$
Initialize B	$B = \begin{cases} \text{pattern-specific} & \text{standard} \\ \frac{1}{5}J_{5\times5} & \text{UNKNOWN} \end{cases}$

Table 2: Phase 0 initialization

4.2 Phase 0: Initialization

4.3 Phase 1: Self Observation with α Update

4.3.1 Mathematical Operations

1. **Self observation**: $V_{\text{self}} \in \mathbb{R}^{5\times25}$

$$V_{\text{self}}[i,:] = \mathrm{filter}(X[i,:], \text{DOM observation})$$

2. **87D transformation**:

$$T_{\text{self}} = T(V_{\text{self}}) \in \mathbb{R}^{5\times87}$$
$$S = T(X) \in \mathbb{R}^{5\times87}$$

3. **Covariance**:

$$S^* = S \cdot T_{\text{self}}^T \in \mathbb{R}^{5\times5}$$

4. **Eigen decomposition**:

$$(\alpha, v_\alpha) = \mathrm{eig}(S^*)$$

5. **Bias update**:

$$b_{\text{initial}} \leftarrow \mathrm{normalize}(\alpha \cdot (v_\alpha v_\alpha^T) \cdot b_{\text{initial}})$$

4.3.2 Example: α Update

Suppose after observation:

$$S^* = \begin{bmatrix} 0.8 & 0.1 & 0.05 & 0.03 & 0.02 \\ 0.1 & 0.7 & 0.1 & 0.05 & 0.05 \\ 0.05 & 0.1 & 0.6 & 0.15 & 0.1 \\ 0.03 & 0.05 & 0.15 & 0.5 & 0.27 \\ 0.02 & 0.05 & 0.1 & 0.27 & 0.56 \end{bmatrix}$$

Eigen decomposition yields:

$$\alpha = 0.85, \quad v_\alpha = [0.6, 0.4, 0.2, -0.1, -0.1]^T$$

Update:

$$b_{\text{initial}} = \mathrm{normalize}(0.85 \times (v_\alpha v_\alpha^T) \times [0.2, 0.2, 0.2, 0.2, 0.2]^T)$$

4.4 UNKNOWN-specific γ Update

Algorithm 1 γ Update for UNKNOWN Pattern

```
procedure GammaUpdate
    Build T_γ tensor (5 × 5 × 87)                          ▷ With void reroutes
    Flatten: G_γ ∈ ℝ^{5×435}
    Covariance: C_γ = G_γ · G_γ^T ∈ ℝ^{5×5}
    Eigen: (γ, v_γ) = eig(C_γ)
    Update: B ← normalize(γ · (v_γ v_γ^T) · B)
end procedure
```

4.5 Phase 2: Competitive Assignment $H(B)$

4.5.1 Hierarchical Selection

For each column $j \in \{0, \ldots, 4\}$:

1. Find $i = \arg\max_i B[i, j]$
2. If ties: $i = \arg\max_{i \in \text{candidates}}(X[i] \cdot V_{\text{self}})$
3. Remove selected i from available rows

Result: indices $I = [i_0, i_1, i_2, i_3, i_4]$

4.5.2 Permutation Transform Y

$$P_{i,k} = \text{PermutationMatrix}(I) \cdot P_i$$

where $\text{PermutationMatrix}(I)[j, i_j] = 1$.

4.5.3 Example: Competitive Assignment

Given B matrix:

$$B = \begin{bmatrix} 0.3 & 0.1 & 0.2 & 0.2 & 0.2 \\ 0.2 & 0.4 & 0.1 & 0.2 & 0.1 \\ 0.1 & 0.2 & 0.5 & 0.1 & 0.1 \\ 0.2 & 0.1 & 0.1 & 0.4 & 0.2 \\ 0.2 & 0.2 & 0.1 & 0.1 & 0.4 \end{bmatrix}$$

$H(B)$ selection:

- Column 0: row 0 (0.3)
- Column 1: row 1 (0.4)
- Column 2: row 2 (0.5)
- Column 3: row 3 (0.4)
- Column 4: row 4 (0.4)

Result: $I = [0, 1, 2, 3, 4]$

4.6 Phase 3: Neighbor Observation with Void Handling

4.6.1 Observation with Membrane Rerouting

For each position $p \in \{\text{parent, up, down, left, right}\}$:

if position has active reroute **then**
    observe(reroute_coordinate)
else if waiting in membrane_waiting **then**
    check membrane system
else
    observe(original_coordinate)
end if

4.6.2 Example: Void Handling Scenario

Parameter	**Value**
Neuron coordinate	(0,1,2)
Void at position	"up"
Void coordinate	(0,1,1)
Candidates found	(0,1,0), (0,0,1), (0,2,1), (0,1,3)
T-similarities	0.85, 0.72, 0.45, 0.63
Selected reroute	(0,1,0)

4.7 Phase 4: Matrix Updates with β Update

4.7.1 Matrix Operations Sequence

Step	**Mathematical Operation**
1	$P_{i,k}^{87D} = T(P_{i,k}) \in \mathbb{R}^{5\times 87}$
2	$W_p^{87D} = T(O) \in \mathbb{R}^{5\times 87}$
3	$D = P_{i,k}^{87D} \cdot (W_p^{87D})^T \in \mathbb{R}^{5\times 5}$
4	$\hat{B} = D \cdot B$
5	$B^* = \text{row\_normalize}(\hat{B})$
6	$(\beta, v_\beta) = \text{eig}(B^*)$
7	$b_{\text{final}} = \text{normalize}(\beta \cdot (v_\beta v_\beta^T) \cdot b_{\text{initial}})$

4.7.2 Example: Matrix Update Computation

Given observations:

$$O = \begin{bmatrix} 0.8 & 0.2 & \cdots & 0.1 \\ 0.1 & 0.9 & \cdots & 0.2 \\ 0.3 & 0.1 & \cdots & 0.8 \\ 0.2 & 0.7 & \cdots & 0.1 \\ 0.6 & 0.3 & \cdots & 0.4 \end{bmatrix}$$

After T-transform:

$$D = \begin{bmatrix} 0.7 & 0.1 & 0.05 & 0.1 & 0.05 \\ 0.05 & 0.8 & 0.05 & 0.05 & 0.05 \\ 0.1 & 0.1 & 0.6 & 0.1 & 0.1 \\ 0.05 & 0.05 & 0.2 & 0.6 & 0.1 \\ 0.1 & 0.1 & 0.1 & 0.1 & 0.6 \end{bmatrix}$$

β update: $\beta = 0.78$, update b_{final}

4.8 Phase 5: Confidence Decision

Let $p_{\text{current}} = \arg\max(b_{\text{final}})$.

$$\text{decision} = \begin{cases} \text{RECYCLING} & \text{if } p_{\text{current}} = \text{current pattern} \\ \text{TENSOR\_FALLBACK} & \text{otherwise} \end{cases}$$

4.8.1 Tensor Fallback with ζ Update

1. Build tensors: $E = T_{\text{exp}}$, $O_{\text{obs}} = T_\zeta(\text{reroutes})$
2. Flatten: $E_f \in \mathbb{R}^{5\times522}$, $O_f \in \mathbb{R}^{5\times522}$
3. Covariance: $G = E_f \cdot O_f^T \in \mathbb{R}^{5\times5}$
4. Eigen: $(\zeta, v_\zeta) = \text{eig}(G)$
5. Update: $b_{\text{grand}} = \text{normalize}(v_\zeta \odot b_{\text{final}})$

5 Eigen Update Sequences and Interpretations

6 Example: Complete Cycle

6.1 Initial Conditions

- Coordinate: (0,1,2)
- Initial pattern: UNKNOWN
- Void at: "up" position (0,1,1)
- Reroute found: (0,1,0)

Pattern	Sequence	Interpretation
Standard patterns	$\alpha \to \beta \to \zeta$	Self $\to$ Position $\to$ Global
UNKNOWN pattern	$\alpha \to \gamma \to \beta \to \zeta$	Self $\to$ Neighbors $\to$ Position $\to$ Global
Eigenvalue	**Range**	**Semantic Meaning**
α	0.0-1.0	Self-identity certainty
γ	0.0-1.0	Neighbor relation consistency
β	0.0-1.0	Position assignment quality
ζ	0.0-1.0	Global pattern consistency

Table 3: Eigen update sequences and interpretations

6.2 Expected Eigenvalues

$$\alpha = 0.92, \quad \gamma = 0.78, \quad \beta = 0.85, \quad \zeta = 0.88$$

6.3 Final Decision

- Final pattern: DATA_INPUT
- Confidence: 0.85
- Active reroutes: 1 ("up" $\to$ (0,1,0))

7 Matrix Dimension Summary

Operation	Input	Output
T transform	$n \times 25$	$n \times 87$
Self covariance	5×87, 5×87	5×5
$H(B)$ selection	5×5	5 indices
Permutation	5×25, 5 indices	5×25
D matrix	5×87, 5×87	5×5
β update	5×5	updated b (5×1)
Tensor fallback	5×522, 5×522	5×5

Table 4: Matrix dimension transformations

8 Normalization Operations

8.1 Vector Normalization

For $v \in \mathbb{R}^n$:

$$\text{normalize}(v)_i = \frac{v_i}{\sum_{j=1}^{n} v_j} \quad \text{with } \epsilon = 10^{-10}$$

8.2 Matrix Row Normalization

For $M \in \mathbb{R}^{m \times n}$:

$$\text{row\_normalize}(M)_{ij} = \frac{M_{ij}}{\sum_{k=1}^{n} M_{ik}}$$

9 System Properties

9.1 Time Complexity

- T transform: $O(n \cdot 25 \cdot 9)$ where $n \leq 5$
- Eigen decomposition: $O(5^3)$ (constant)
- Complete cycle: $O(1)$ bounded operations

9.2 Expected Behavior

- **High confidence**: $> 0.7 \rightarrow$ enter recycling mode
- **Medium confidence**: $0.4 - 0.7 \rightarrow$ continue processing
- **Low confidence**: $< 0.4 \rightarrow$ likely UNKNOWN or pattern switch
- **Void handling**: Automatic rerouting preserves relational patterns
- **UNKNOWN specialization**: γ update refines neighbor expectations