Closing the Alignment Gap: The Resonant State Alignment Algorithm (RSAA) for Deliberative Intelligence

Gianne P. Bacay [A.R.C.A.N.E. Research. Email: giannebacay2004@gmail.com]

Abstract

Traditional feed-forward architectures suffer from a fundamental limitation: information flows in one direction, preventing lower layers from revising their representations based on higher-level context. This creates an **Alignment Gap** between local feature extraction and global semantic coherence. Backpropagation addresses this only retrospectively, adjusting weights *after* errors propagate to the output, but it cannot refine internal states in real-time before a response is committed. To solve this problem, I developed the **Resonant State Alignment Algorithm (RSAA)**, which closes this gap by enabling hierarchical systems to achieve internal coherence *prospectively* through iterative bi-directional state refinement. RSAA decouples state alignment from weight modification: layers first "resonate" to reach mutual consistency, then optionally update weights based on the aligned states. This is the first formalization of a deliberative alignment mechanism that allows computational systems to "think" before responding. The algorithm is substrate-agnostic, applying beyond neural networks to quantum variational circuits, cybernetic control systems, and multi-agent coordination.

---

1. Introduction

In a hierarchical computational system $\mathcal{H}$ with L layers, each layer $i \in \{1, \dots, L\}$ maintains an internal state representation $S_i \in \mathbb{R}^{d_i}$. Traditional neural architectures process information in a single feed-forward pass, mapping inputs to outputs without any opportunity for layers to "re-evaluate" their conclusions based on higher-level context. This unidirectional flow creates a significant **Alignment Gap**: a disconnect between low-level feature extraction and high-level contextual constraints.

The textbook approach to neural learning, backpropagation, combined with advanced optimizers such as Adam or SGD with momentum, solves the credit assignment problem by propagating a global error signal backward through the network. However, this approach is:

1. **Retrospective**: It identifies errors *after* an output is generated.
2. **Global**: It requires a complete forward pass before any correction can occur.
3. **Biologically Implausible**: The brain does not utilize such global error signals.

For biological neural systems, the neocortex is organized into a hierarchy where every feed-forward connection is matched by a feedback connection. Predictive Coding theory suggests that the brain constantly generates top-down predictions about what lower-level inputs *should* be, and learning occurs when these predictions fail to match reality.

1.1 Main Results

In this paper, I present the Resonant State Alignment Algorithm (RSAA), a formal algorithm for closing the Alignment Gap through iterative state refinement, enabling deliberative intelligence in hierarchical systems.

Theorem 1.1. *There exists a deterministic algorithm that takes $O(N \cdot L)$ time to achieve Prospective Configuration in a hierarchy $\mathcal{H}$ of L layers, where N is the number of resonance cycles. The algorithm operates solely on internal state activations $\{S_1, \dots, S_L\}$ without modifying synaptic weights W.*

Note that the algorithm is substrate-agnostic; it applies equally to artificial neural networks, quantum variational circuits, cybernetic control systems, and multi-agent coordination mechanisms.

1.2 Technical Overview

Broadly speaking, there are two traditional paradigms for neural computation:

- **Feed-Forward Inference**: Information flows from input to output in a single pass. This is computationally efficient but lacks deliberative capacity.
- **Backpropagation Learning**: A global error signal propagates backward to adjust weights. This is effective for learning but does not refine activations in real-time.

My approach merges these two paradigms through a **Resonance Loop** technique. At any point during the execution of RSAA, each layer maintains a "state" S_i that represents its current interpretation of the input. A higher layer i can "project" its expectation $P_{i \to i-1}$ to the layer below, and the lower layer $i-1$ can "harmonize" its state to reduce the divergence $\Delta_{i-1} = S_{i-1} - P_{i \to i-1}$.

The key insight is that by iterating this projection-harmonization cycle N times *before* any weight update or output commitment, the hierarchy reaches an equilibrium state where all layers are mutually consistent. I call this equilibrium **Resonance**.

My most essential idea is the separation of **state alignment** from **weight modification**. Traditional learning conflates these two processes: the only way to reduce error is to change weights. RSAA decouples them: states are first aligned (Prospective Configuration), and only then are weights optionally updated based on the aligned states. This yields several benefits:

- **Stability**: Local corrections prevent gradient explosion/vanishing.
- **Deliberation**: The system can "think" before committing to an output.
- **Biological Plausibility**: The mechanism mirrors neocortical feedback loops.

---

2. Preliminaries

Definition 2.1 (Resonant Hierarchy). A Resonant Hierarchy $\mathcal{H}$ is an ordered set of L layers, where each layer i maintains:

- An internal state $S_i \in \mathbb{R}^{d_i}$.
- A projection function $f_{proj}^{(i)} : \mathbb{R}^{d_i} \to \mathbb{R}^{d_{i-1}}$.
- A harmonization rate $\gamma_i \in (0, 1]$.

Definition 2.2 (Feedback Projection). For a layer i, the Feedback Projection to layer $i-1$ is:

$$P_{i \to i-1} = f_{proj}^{(i)}(S_i; W_{i,proj})$$

In the A.R.C.A.N.E. implementation, this is typically the matrix transpose of the input weights:

$$P_{i \to i-1} = S_i \cdot W_i^T$$

Definition 2.3 (Prediction Divergence). The Prediction Divergence at layer $i-1$ is the signed difference between its current state and the expectation projected from above:

$$\Delta_{i-1} = S_{i-1} - P_{i \to i-1}$$

Definition 2.4 (Global Divergence). The Global Divergence $\mathcal{D}$ of a hierarchy $\mathcal{H}$ is the sum of squared local divergences:

$$\mathcal{D} = \sum_{i=1}^{L-1} \|\Delta_i\|^2$$

Definition 2.5 (Resonance). A hierarchy $\mathcal{H}$ is said to be in Resonance when $\mathcal{D} < \epsilon$ for some threshold $\epsilon > 0$.

3. The RSAA Algorithm

3.1 State Harmonization

The core operation of RSAA is the State Harmonization update rule.

Lemma 3.1 (Harmonization Update). *Given a layer $i-1$ with state $S_{i-1}^{(t)}$ and an incoming projection $P_{i\to i-1}$, the harmonized state at cycle $t+1$ is:*

$$S_{i-1}^{(t+1)} = S_{i-1}^{(t)} - \gamma \cdot \Delta_{i-1}^{(t)}$$

where $\gamma \in (0, 1]$ is the Resonance Factor.

Proof. The update directly minimizes the local divergence $\|\Delta_{i-1}\|^2$ via gradient descent on the state variable S_{i-1}. Since $\Delta_{i-1} = S_{i-1} - P_{i\to i-1}$, the gradient with respect to S_{i-1} is $\nabla_{S_{i-1}} \|\Delta_{i-1}\|^2 = 2\Delta_{i-1}$. A gradient descent step with learning rate $\gamma/2$ yields the update rule. □

3.2 Convergence

Theorem 3.2 (Convergence of RSAA). *For a Resonant Hierarchy $\mathcal{H}$ with fixed projections $\{P_{i\to i-1}\}$, the RSAA update rule converges to $\mathcal{D} = 0$ as $N \to \infty$, provided $\gamma \in (0, 1]$.*

Proof. At each cycle, the local divergence $\|\Delta_{i-1}^{(t+1)}\|^2 = \|(1-\gamma)\Delta_{i-1}^{(t)}\|^2 = (1-\gamma)^2\|\Delta_{i-1}^{(t)}\|^2$. Since $(1-\gamma)^2 < 1$ for $\gamma \in (0, 1]$, the divergence decreases geometrically. Summing over all layers, $\mathcal{D}^{(t+1)} \leq (1-\gamma)^2\mathcal{D}^{(t)}$, which converges to 0. □

3.3 Algorithmic Flow

Algorithm 1: Resonant State Alignment

```
Input: Hierarchy H, Input x, Cycles N, Threshold epsilon
Output: Aligned states {S_1, ..., S_L}

1.  [Forward Initialization]
    Perform feed-forward pass to populate {S_1, ..., S_L}.

2.  [Resonance Loop]
    for t = 1 to N do:
        // Step A: Project (Top-Down)
        for i = L down to 2 do:
            P_{i -> i-1} = f_proj(S_i; W_i)
        end for

        // Step B: Harmonize (Bottom-Up)
        for i = 1 to L-1 do:
            Delta_i = S_i - P_{i+1 -> i}
            S_i = S_i - gamma * Delta_i
        end for

        // Step C: Check Convergence
        D = sum of ||Delta_i||^2
        if D < epsilon then break
    end for
```

```
3.  [Final Inference]
    Return output Y = f_output(S_L)
```

---

4. Substrate Agnosticism

A key property of RSAA is that it operates on abstract principles (state-space negotiation, local correction dynamics, stability-driven refinement) rather than assumptions unique to artificial neural networks.

Corollary 4.1. *RSAA can be instantiated in any system $\mathcal{S}$ that satisfies:*

1. *$\mathcal{S}$ maintains adjustable internal states $\{S_i\}$.*
2. *$\mathcal{S}$ supports a projection operation between state levels.*
3. *$\mathcal{S}$ supports an additive update operation on states.*

This includes:

- **Quantum Computing**: Variational state stabilization in VQE/QAOA circuits.
- **Cybernetics**: Maintaining equilibrium in feedback-rich physical systems.
- **Multi-Agent Systems**: Achieving collective agreement through decentralized resonance.
- **Symbolic AI**: Enforcing semantic consistency between logical representations.

---

5. Conclusion

The Resonant State Alignment Algorithm (RSAA) closes the Alignment Gap by providing a rigorous mathematical basis for deliberative intelligence. By treating alignment as a dynamic convergence process rather than a static mapping, RSAA allows computational systems to achieve internal coherence before committing to an output. The decoupling of state alignment from weight modification represents a paradigm shift from purely retrospective learning toward prospective, self-modeling computation. This work demonstrates that deliberative intelligence, the capacity to "think" before responding, can be formally achieved through iterative resonant state refinement.

References

- [ART87] Carpenter, G. A., & Grossberg, S. (1987). A massively parallel architecture for a self-organizing neural pattern recognition machine. *Computer Vision, Graphics, and Image Processing*, 37(1), 54-115.
- [PC99] Rao, R. P., & Ballard, D. H. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2(1), 79-87.
- [FEP10] Friston, K. (2010). The free-energy principle: a unified brain theory? *Nature Reviews Neuroscience*, 11(2), 127-138.
- [PC21] Millidge, B., Tschantz, A., & Buckley, C. L. (2021). Predictive Coding Approximates Backprop along Arbitrary Computation Graphs. *Neural Computation*, 34(6), 1329-1368.
- [HEB49] Hebb, D. O. (1949). *The Organization of Behavior*. Wiley.
- [BP86] Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088), 533-536.