

Research Proposal: Exploring Chiral Topologies for Enhanced Gradient Descent

Paul Lowndes
ZeroTrust@NSHkr.com

December 4, 2024

Abstract

This research proposal outlines a novel approach to gradient descent optimization, termed Chiral Gradient Descent (CGD), which incorporates topological information and rotational dynamics inspired by chirality in biological systems. The proposed methodology will investigate how chiral structures within neural networks can be leveraged to improve exploration of the parameter space, potentially leading to more robust and efficient training. The research will involve developing a mathematical framework for CGD, implementing the algorithm in a deep learning framework, and conducting experiments on benchmark datasets to evaluate CGD's performance compared to standard gradient descent methods. The expected outcomes include a refined mathematical formulation of CGD, an open-source implementation, and experimental results demonstrating the potential benefits of CGD for various machine learning tasks.

1 Introduction

Gradient descent is widely used for optimizing machine learning models, but its susceptibility to local minima remains a challenge. Chiral gradient descent (CGD) presents a novel approach, drawing inspiration from the concept of chirality. CGD aims to enhance standard gradient descent by introducing asymmetry into the update process, motivated by the observation that many biological systems, including neural networks in the brain, exhibit chiral structures. We hypothesize this could lead to more efficient exploration of the parameter space and faster, more stable convergence.

Despite the widespread use of gradient descent, its limitations in navigating complex, non-convex landscapes have prompted the exploration of alternative methods. Chiral Gradient Descent (CGD) introduces a novel mechanism by leveraging chiral asymmetry, inspired by natural systems where chirality plays a crucial role in dynamic interactions. By incorporating topological information and rotational dynamics, CGD aims to enhance the exploration capabilities of traditional gradient descent, potentially leading to faster and more robust convergence. This builds on previous work exploring various gradient descent variants [1].

2 Chiral Gradient Descent

Chiral Gradient Descent (CGD) modifies the gradient update rule by incorporating chiral vectors, which introduce rotational dynamics into the optimization process. This approach is inspired by natural asymmetry observed in biological systems. The chiral term, incorporating a sigmoid function, allows for dynamic adjustments based on topological distances within the network, potentially enhancing exploration of the parameter space and leading to more robust convergence. This sigmoid function modulates the influence of each chiral pair based on the topological distance between neurons, allowing for local chiral effects to dominate while diminishing the impact of distant pairs. The mathematical formulation of CGD involves the cross product of the gradient with a chiral vector, adding a layer of complexity and potential to the optimization process.

3 Applying Chiral Topologies

We represent a neural network’s topology as a graph $G = (V, E)$. A chiral pair of neurons (v_i, v_j) is defined based on topological asymmetry, such as differences in shortest path lengths from a common ancestor. For each chiral pair, a chiral vector \mathbf{c}_{ij} is defined, reflecting their topological relationship. Measures such as differences in path lengths, or other metrics relating to properties of chiral topologies will determine a numerical representation for relative asymmetry and help refine our distance metric between them. These metrics could involve other topological or information-theoretic properties like local or global curvature or the orientation or density of node or edge data.

4 Mathematical Formulation

The core innovation of CGD lies in its gradient calculation which incorporates chiral vectors:

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha \nabla L(\boldsymbol{\theta}_t) + \beta \sum_{i,j \in C} s(w_{ij}, \mathbf{c}_{ij})(\nabla L(\boldsymbol{\theta}_t) \times \mathbf{c}_{ij}) \quad (1)$$

Where:

- $\boldsymbol{\theta}_t$ represents the parameter vector at iteration t .
- α denotes the learning rate.
- $\nabla L(\boldsymbol{\theta}_t)$ is the gradient of the loss function at iteration t .
- β represents the chirality parameter, which modulates the influence of the chiral vectors.

- C denotes the set of chiral pairs being considered during the update step, which may vary at each iteration depending on the method or constraints being used by the researcher.
- w_{ij} represents a weight associated with the chiral pair (i, j) , and may reflect asymmetry measures related to properties of their chiral topologies.
- \mathbf{c}_{ij} represents the chiral vector for the chiral pair (i, j) .
- \times represents the cross-product.
- $s(w_{ij}, \mathbf{c}_{ij})$ is a function designed to blend the chiral vector's influence with considerations based on a weight w_{ij} .

5 Training Intuition and Higher Dimensions

Understanding chirality in higher dimensions can begin with visualizing simple cases (2D and 3D), followed by mathematical generalization to 4D, 5D, and beyond. Training involves visualization, representation with vectors and matrices, implementing simple transformations, and generalizing to higher dimensions.

Understanding the role of chirality in higher dimensions requires a shift in perspective, as traditional geometric intuitions may not directly apply. By visualizing lower-dimensional cases and gradually extending these insights to higher dimensions, researchers can develop a deeper intuition for the impact of chiral dynamics on learning. The use of vector and matrix representations allows for the implementation of simple transformations that can be generalized, providing a framework for exploring the effects of chirality across various dimensional spaces.

6 Towards CGD: A Synthesis

The central idea of Chiral Gradient Descent (CGD) is to enhance standard gradient descent by introducing chirality, a property of asymmetry, into the update process. This is motivated by the prevalence of chiral structures and behaviors in many biological systems. A CGD equation, incorporating a sigmoid function is presented below:

Here's a new section for your research paper, incorporating the method from the provided paper ("Machine Learning Topological Invariants with Neural Networks") for identifying chiral pairs and adapting it to your needs. This will form Phase 1 of your system.

7 Identifying Chiral Pairs: A Topologically-Informed Approach

This section details a novel method for identifying chiral pairs within complex networks, extending the approach described in Zhang et al. [2] for identifying topological invariants. Instead of directly predicting topological invariants, we adapt their convolutional neural network (CNN) architecture to identify pairs of nodes exhibiting chiral topological features, focusing on asymmetries within the network’s structure and information flow. This will form Phase 1 of our system for identifying chiral pairs to be used in subsequent phases to implement chiral gradient descent.

7.1 Network Representation

As in the previous sections, we represent networks as directed graphs $G = (V, E)$, where V is the set of nodes and E is the set of directed edges. Each edge $e_{ij} \in E$ connecting node v_i to node v_j has an associated weight w_{ij} representing the strength of the connection (e.g., correlation between node activations, information flow, or interaction strength). We extend the graph representation by including additional node attributes that might influence the identification of chiral pairs, such as node centrality, community membership, and other topological measures that could prove useful in identifying pairs in the network. This extended representation is richer and more nuanced compared to simple directed graphs and is necessary to capture the more complex relationships between nodes in the network.

As in the previous sections, we represent networks as directed graphs $G = (V, E)$, where V is the set of nodes and E is the set of directed edges. Each edge $e_{ij} \in E$ connecting node v_i to node v_j .

7.2 Adapting the Convolutional Neural Network

We adapt the CNN architecture proposed in Zhang et al. [2] (see Figure 1 in the original paper) to learn local topological features related to chirality. The input to the CNN will be a matrix representation of the local neighborhood around each node in the graph. This representation will be constructed by including several elements:

Node Attributes: Include node attributes such as centrality and community membership in the input matrix. **Edge Weights:** The edge weights from the node to its neighbors are added to the input matrix. **Shortest Path Lengths:** Compute the shortest path lengths between each pair of neighbors, which will inform the computation of the chiral vector in the subsequent phases.

This extended input representation captures both local topology and more global network features. This information is crucial for accurately identifying chiral pairs, unlike the approach in the original paper which only focused on computing the winding number.

7.3 Chiral Pair Identification

The output of the CNN is a vector that represents a topological feature embedding of the local neighborhood for each node. We define a chiral pair as a pair of nodes whose topological feature embeddings show a high degree of asymmetry or anti-correlation. This asymmetry or anti-correlation is evaluated using a distance metric, such as cosine similarity. We select the top pairs that maximize the asymmetry as the chiral pairs relevant to performing CGD in subsequent phases. The selection process could be made more complex to filter out pairs with specific traits or incorporate more data to reduce computational cost or improve performance.

7.4 Hierarchical Application

To handle hierarchical structures, we apply the method recursively to sub-graphs within the network. This hierarchical application identifies chiral pairs at different scales, allowing for a more granular analysis of asymmetry in multi-level structures, which mirrors the multi-scale nature of hierarchical structures in narratives and social networks.

7.5 Algorithm

Algorithm 1 Chiral Pair Identification

Require: Graph $G = (V, E)$, CNN model, asymmetry threshold τ

```
1: Initialize empty set of chiral pairs  $C$ 
2: for each node  $v_i \in V$  do
3:   Construct input matrix  $M_i$  for  $v_i$ 's neighborhood
4:   Obtain feature embedding vector  $F_i$  from  $\text{CNN}(M_i)$ 
5: end for
6: for each pair of nodes  $(v_i, v_j) \in V \times V$  do
7:   Compute asymmetry score  $A_{ij} = d(F_i, F_j)$  (e.g., using cosine distance)
8:   if  $A_{ij} > \tau$  then
9:     Add  $(v_i, v_j)$  to  $C$ 
10:  end if
11: end for
12: return Set of chiral pairs  $C$ 
```

7.6 Discussion

This method builds upon the success of CNNs in learning complex patterns from local data and provides a rigorous approach to identify chiral pairs. By incorporating topological features and hierarchical application, it addresses the complexities of analyzing asymmetries in real-world systems. The choice of CNN architecture and asymmetry threshold requires careful evaluation and tuning for specific applications and datasets.

We will compare this approach with alternative methods for measuring graph similarity and asymmetry in our validation experiments.

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha \nabla L(\boldsymbol{\theta}_t) + \beta \sum_{i,j \in C(\boldsymbol{\theta}_t)} \frac{\|\mathbf{c}_{ij}\|}{1 + e^{-\gamma d_{ij}}} (\nabla L(\boldsymbol{\theta}_t) \times \mathbf{c}_{ij}) \quad (2)$$

Where:

- $\boldsymbol{\theta}_t$: Parameter vector at time t .
- α : Learning rate.
- $\nabla L(\boldsymbol{\theta}_t)$: Gradient of the loss function.
- β : Global chirality parameter.
- $C(\boldsymbol{\theta}_t)$: Set of relevant chiral pairs, potentially changing dynamically during training.
- \mathbf{c}_{ij} : Chiral vector for pair (i, j) , calculated based on topology.
- d_{ij} : Topological distance between i and j based on features like difference in path lengths, curvature measures, node or edge distribution densities, etc.
- γ : Scaling parameter for the sigmoid function, influencing the impact of d_{ij} .

Algorithm 2 Chiral Gradient Descent (CGD)

Require: Learning rate α , chirality parameter β , scaling parameter γ , initial parameters $\boldsymbol{\theta}_0$

while not converged **do**

 Compute gradient: $\nabla L(\boldsymbol{\theta}_t)$

 Determine relevant chiral pairs: $C(\boldsymbol{\theta}_t)$

 Calculate chiral vectors \mathbf{c}_{ij} and distances d_{ij} for $(i, j) \in C(\boldsymbol{\theta}_t)$

$\Delta \boldsymbol{\theta} = \beta \sum_{i,j \in C(\boldsymbol{\theta}_t)} \frac{\|\mathbf{c}_{ij}\|}{1 + e^{-\gamma d_{ij}}} (\nabla L(\boldsymbol{\theta}_t) \times \mathbf{c}_{ij})$

 Update parameters: $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \alpha \nabla L(\boldsymbol{\theta}_t) + \Delta \boldsymbol{\theta}$

$t \leftarrow t + 1$

end while

This refined CGD algorithm uses a sigmoid function to modulate the influence of each chiral pair based on the topological distance d_{ij} between neurons. Chiral pairs that are topologically "closer" (smaller d_{ij}) have a stronger influence on the update. The magnitude of the chiral vector $\|\mathbf{c}_{ij}\|$ also contributes, allowing pairs with greater asymmetry to exert more influence. The parameter β controls the global effect of chirality, while γ modulates the sigmoid's steepness, providing control over the sensitivity to topological distances d_{ij} .

8 Identifying Chiral Pairs: A Topologically-Informed Approach

This section details a method for identifying chiral pairs, building upon Zhang *et al.* [1] but incorporating novel elements to capture asymmetry relevant to narrative structures and social networks. This forms Phase 1 of our system, providing the foundation for chiral gradient descent.

8.1 Network Representation (Enhanced)

We represent networks as directed graphs $G = (V, E)$ with weighted edges w_{ij} . Crucially, we augment this with **relative path information**. For each node v_i , we pre-compute:

- **Shortest Path Lengths:** The shortest path length from v_i to every other node v_j (denoted $d(v_i, v_j)$).
- **Common Ancestors and Path Differences:** For each pair of nodes (v_i, v_j) , we identify their common ancestors and calculate the *difference* in path lengths from each common ancestor to v_i and v_j . This captures the asymmetry in how v_i and v_j relate to their shared history or context.

This richer representation goes beyond simple connectivity, encoding structural asymmetries relevant to narratives (e.g., how different characters relate to key events) and social networks (e.g., differing perspectives on shared information).

8.2 Convolutional Neural Network (Adapted)

We adapt the CNN architecture from Zhang *et al.* [1] to learn chiral features. The input to the CNN for node v_i now includes:

- **Local Neighborhood Structure:** Edge weights w_{ij} for v_j in v_i 's neighborhood.
- **Relative Path Information:** For each neighbor v_j , include the shortest path length $d(v_i, v_j)$ and the differences in path lengths from common ancestors.

This combined input allows the CNN to learn features sensitive to both local connectivity and global topological asymmetries.

8.3 Chiral Pair Identification (Refined)

Instead of simply using cosine distance, we define a more nuanced chirality score:

$$\text{ChiralScore}(v_i, v_j) = \text{Asymmetry}(F_i, F_j) \times \text{PathDifference}(v_i, v_j) \quad (3)$$

Where:

- F_i, F_j are the feature embeddings from the CNN for nodes v_i and v_j .
- $\text{Asymmetry}(F_i, F_j)$ measures the asymmetry between the embeddings (e.g., using cosine distance or a learned metric).
- $\text{PathDifference}(v_i, v_j)$ is a weighted average of the path length differences from common ancestors, emphasizing structural asymmetry.

Pairs with high ChiralScore are identified as chiral pairs. This combined score captures both feature-level and structural asymmetry.

8.4 Hierarchical Application (Maintained)

The hierarchical application from the previous version is maintained, applying the method recursively to identify chiral pairs at different scales.

8.5 Algorithm (Refined LaTeX)

Algorithm 3 Chiral Pair Identification (Refined)

Require: Graph $G = (V, E)$, CNN model, chirality threshold τ

- 1: Pre-compute shortest path lengths and path differences for all node pairs
 - 2: Initialize empty set of chiral pairs C
 - 3: **for** each node $v_i \in V$ **do**
 - 4: Construct input matrix M_i (neighborhood, paths)
 - 5: $F_i \leftarrow \text{CNN}(M_i)$
 - 6: **end for**
 - 7: **for** each pair of nodes $(v_i, v_j) \in V \times V$ **do**
 - 8: $S_{ij} \leftarrow \text{ChiralScore}(v_i, v_j)$
 - 9: **if** $S_{ij} > \tau$ **then**
 - 10: $C \leftarrow C \cup \{(v_i, v_j)\}$
 - 11: **end if**
 - 12: **end for** **return** C
-

8.6 Discussion (Expanded)

This refined method directly addresses the challenge of defining and detecting chirality relevant to narrative and social network analysis. By incorporating relative path information and a combined chirality score, it captures more nuanced asymmetries than simply comparing feature embeddings. The use of common ancestor paths adds a "historical" or "contextual" dimension to the chirality measure, which is particularly relevant for understanding how information and relationships evolve in narratives and social networks. The hierarchical application and the flexibility in defining the *Asymmetry* and *PathDifference* functions provide adaptability for various applications.

9 Conclusion

This research proposal presents a novel approach to gradient descent optimization that holds significant promise. By leveraging the power of chiral topologies and incorporating biologically plausible mechanisms into the optimization process, this research has the potential to overcome the limitations of traditional gradient descent and usher in a new era of more efficient and effective deep learning models. The research plan detailed above, if successfully executed, will provide valuable insight into the use of chiral gradient descent and pave the way for its deployment in real-world applications. The next steps will involve developing and testing the CGD algorithm, conducting rigorous experiments, and analyzing the findings to validate its performance and contribute to the advancement of deep learning methodologies.

References

- [1] Induraj. Different variants of gradient descent. <https://induraj2020.medium.com/different-variants-of-gradient-descent-6854d3118c4c>, 2023. Accessed: 2024-12-04.
- [2] P. Zhang, H. Shen, and H. Zhai. Machine learning topological invariants with neural networks. *Physical Review B*, 97(20):205138, 2018.