Visualizing Ambiguity: A Grammar of Graphics Approach to Resolving Numerical Ties in Parallel Coordinate Plots

Comprehensive Exam Summary

Denise Bradford

2025-11-01

Table of contents

1 Executive Summary

This document summarizes my research on resolving numerical ties in parallel coordinate plots (PCPs). The research addresses a fundamental limitation in multivariate data visualization: when multiple observations share identical numerical values, they create perfectly overlapping polylines that obscure density information and substructure. This work develops three deterministic jittering algorithms—Sunflower, Halton, and Intelligent—that systematically resolve numerical ties while maintaining reproducibility and perceptual validity.

Key Contributions:

- Development of three novel deterministic jittering algorithms for numerical tie resolution

- Integration with the ggpcp R package within a grammar of graphics framework
- Theoretical analysis comparing biomimetic and quasi-random distribution methods
- Comprehensive user study design for empirical evaluation
- Documentation of a negative result (Intelligent jitter) with valuable design insights

2 Introduction

2.1 Background and Motivation

Parallel Coordinate Plots (PCPs) map n-dimensional data onto two-dimensional displays using parallel vertical axes (Inselberg 1985; Wegman 1990). These visualizations are fundamental tools for:

- High-dimensional data exploration
- Cluster detection
- Pattern recognition
- Multivariate relationship analysis

However, PCPs face a critical limitation: **visual occlusion from numerical ties**.

2.2 The Core Problem

When multiple observations share identical numerical values across dimensions:

1. **Visual Collision**: Polylines stack perfectly on identical paths
2. **Density Information Loss**: A single visible line may represent 1 or 1,000 observations
3. **Structural Occlusion**: Sub-clusters and outliers within tied groups become invisible

This violates a fundamental principle of statistical graphics: visual magnitude should correspond to data magnitude (Cleveland and McGill 1984).

2.3 Research Context

2.3.1 Historical Development

- **1985**: Inselberg introduces parallel coordinates for geometric duality and pattern recognition
- **1990**: Wegman applies PCPs to statistical data analysis
- **1990s-2000s**: Recognition of overplotting problem (Wegman and Luo 1991; Johansson and Forsell 2016)

- **2006**: Parallel Sets elegantly solve categorical data visualization (Kosara, Bendix, and Hauser 2006)
- **2021**: Quality metrics for parallel sets developed (Dennig et al. 2021)
- **2023**: ggpcp package introduces grammar of graphics framework (VanderPlas et al. 2023)

2.3.2 The Persistent Gap

While categorical tie-breaking has been systematically solved, numerical ties have remained without formal treatment—until now.

3 The Problem in Detail

3.1 Three Dimensions of Occlusion

3.1.1 1. Visual Collision

Multiple polylines overlay exactly, making it impossible to distinguish:

- Whether a path represents 1 or 100 observations
- The relative frequency of different patterns
- The density distribution within tied groups

3.1.2 2. Density Information Loss

Without separation, analysts cannot:

- Estimate the number of observations following a particular path
- Identify which patterns are common vs. rare
- Make valid statistical inferences about frequency distributions

3.1.3 3. Structural Occlusion

Hidden information includes:

- Sub-clusters within tied value groups
- Outliers that deviate on other dimensions
- Multivariate patterns that require line tracing (Johansson et al. 2005)

3.2 Concrete Example: Iris Dataset

The classic Iris dataset demonstrates this problem clearly:

- 150 observations across 4 continuous dimensions
- Integer and half-integer measurements create natural ties
- Species-specific patterns masked by overlapping lines

In standard PCPs of the Iris data, severe occlusion makes it difficult to:

- Distinguish the three species
- Estimate group sizes
- Identify within-species variation
- Trace individual observation paths

4 Design Requirements

Any solution to numerical tie resolution must satisfy four core principles:

4.1 1. Determinism

Requirement: Identical input must produce identical output.

Rationale: Essential for scientific reproducibility (Peng 2011). Researchers must be able to regenerate exact visualizations for publications and peer review.

Implication: Rules out standard random jitter approaches.

4.2 2. Uniformity

Requirement: Even distribution within displacement interval with minimal clustering.

Rationale:

- Faithful representation of density
- Minimize artificial patterns
- Ensure visual density corresponds to data frequency

Implication: Random methods create incidental clusters due to the birthday paradox effect.

4.3 3. Perceptual Validity

Requirement: Displacement must balance two competing needs:

- Small enough to maintain value integrity
- Large enough for visual separation

Rationale: Users must be able to trust that displaced values remain close to true values while still being visually distinguishable (Ware 2012).

Implication: Requires careful parameter selection and potentially adaptive methods.

4.4 4. Scalability

Requirement: Handle 2 to 10,000+ observations per tie group efficiently.

Rationale:

- Real-world datasets vary enormously in size
- Interactive exploration requires real-time performance
- Computational efficiency enables integration into standard workflows

Implication: Algorithms must have favorable asymptotic complexity.

5 Methodology

5.1 Mathematical Framework

For each tied value v with n_{ties} observations, we distribute points within a displacement interval:

$$\left[v - \frac{\epsilon}{2}, v + \frac{\epsilon}{2}\right]$$

where ϵ is the maximum displacement magnitude, typically 0.05–0.10 of the axis range.

5.1.1 Optimization Goals

1. Maximize minimum inter-point distance
2. Minimize visual artifacts (clustering, patterns)
3. Maintain deterministic reproducibility
4. Achieve uniform coverage of the interval

5.2 Three Deterministic Algorithms

All three methods provide:

- Deterministic, reproducible output
- Theoretically-grounded distributions
- Computational efficiency ($O(n)$ per tie group)

5.2.1 Algorithm Summary

Algorithm	Theoretical Basis	Key Property	Dimension
Sunflower	Phyllotaxis (Vogel 1979)	Biomimetic optimal packing	2D $\rightarrow$ 1D
Halton	Quasi-random sequences (Halton 1960)	Low-discrepancy guarantees	Pure 1D
Intelligent	Golden ratio application	Linear progressive reveal	Hybrid 1D

6 Sunflower Jitter: Biomimetic Approach

6.1 Biological Inspiration

Sunflower seeds arrange themselves following nature's optimization solution for packing efficiency (Vogel 1979):

The Golden Angle: $137.508° = 360° \times (2 - \phi)$

where $\phi = \frac{1+\sqrt{5}}{2}$ is the golden ratio.

6.1.1 Why This Angle?

The golden angle is the "most irrational" number in the continued fraction sense, meaning:

- It has the slowest-converging continued fraction expansion
- It is maximally incommensurable with full rotations
- Seeds never align along the same radius, even after hundreds of iterations

This property has been optimized by evolution over millions of years to achieve near-perfect packing density.

6.2 Mathematical Formulation

For observation j in a tie group of size n_{ties}:

$$\text{angle}_j = (j-1) \times 137.508°$$

$$\text{radius}_j = \epsilon \times \sqrt{\frac{j-1}{n_{\text{ties}}}}$$

$$\text{displacement}_j = \text{radius}_j \times \cos(\text{angle}_j)$$

6.2.1 Key Features

Square Root Scaling: Maintains constant density as radius increases. In a 2D disk:

- Circumference at radius r: $2\pi r$
- Number of points at radius r_j: proportional to j
- For constant density: $r \propto \sqrt{j}$ balances linear growth in points with radial expansion

Cosine Projection: Maps 2D polar coordinates to 1D linear displacement while preserving distribution properties.

Spiral Structure: The characteristic spiral pattern is preserved even in 1D projection.

6.3 Theoretical Properties

- Near-optimal minimum separation (Vogel 1979)
- Low-discrepancy properties in 2D
- Aesthetically consistent patterns
- Evolutionarily validated optimization

6.4 Visual Results

When applied to the Iris dataset, Sunflower jitter:

- Clearly reveals density differences between species
- Maintains species clustering patterns
- Enables individual line tracing
- Creates aesthetically pleasing, organic-looking distributions

7 Halton Jitter: Quasi-Random Sequences

7.1 Beyond Pseudo-Randomness

Halton sequences (Halton 1960) are:

- **Deterministic** (not random)
- **Low-discrepancy** (fill space uniformly)
- **Number-theoretically constructed** using prime bases

7.1.1 The Random Number Problem

Pseudo-random numbers inevitably cluster due to chance (birthday paradox):

- In 100 random points, gaps and clusters are statistically inevitable
- Visual artifacts mislead analysts
- Density perception becomes unreliable

7.1.2 Halton's Solution

Place each new point as far as possible from all previous points through systematic construction using the **van der Corput sequence**.

7.2 Construction Method

The van der Corput sequence in base 2:

i	Binary	Reversed	Decimal h_i
0	0	0	0.0
1	1	1	0.5
2	10	01	0.25
3	11	11	0.75
4	100	001	0.125
5	101	101	0.625

Pattern: Each point bisects the largest remaining gap.

7.2.1 Application to PCPs

$$\text{displacement}_i = \epsilon \times (h_i - 0.5)$$

Centering around 0.5 creates symmetric bidirectional displacement.

7.3 Theoretical Guarantees

7.3.1 Discrepancy Theory (Niederreiter 1992)

Measure of how uniformly points fill an interval:

$$D_n = \sup_{I \subseteq [0,1]} \left| \frac{\#\{x_i \in I\}}{n} - |I| \right|$$

Theoretical Bounds:

- Random sequences: $D_n = O(n^{-1/2})$
- Halton sequences: $D_n = O(n^{-1} \log n)$
- Optimal lower bound: $D_n = \Omega(n^{-1} \log n)$

Conclusion: Halton is near-optimal in the information-theoretic sense.

7.4 Applications Beyond PCPs

Halton sequences are widely used in:

- Monte Carlo integration
- Computer graphics sampling
- Quasi-Monte Carlo methods
- Numerical analysis

7.5 Visual Results

When applied to the Iris dataset, Halton jitter:

- Provides uniform distribution with no artificial clustering
- Enables accurate density perception
- Maintains deterministic reproducibility
- Creates visually clean separations

8 Intelligent Jitter: A Novel Approach

8.1 Design Motivation

Research Question: Can we apply the golden ratio directly in 1D rather than through angular spacing?

8.2 Algorithm Design

$$\text{angle}_i = (i-1) \times 2\pi \times 0.618$$

$$\text{displacement}_i = \epsilon \times \cos(\text{angle}_i) \times \frac{i-1}{n_{\text{ties}}}$$

8.2.1 Key Distinctions from Sunflower

- Uses golden ratio **value** (0.618) not golden angle (137.5°)
- **Linear scaling**: $\frac{i-1}{n_{\text{ties}}}$ instead of $\sqrt{i/n}$
- 1D modulation rather than 2D-to-1D projection

8.3 Hypothesis vs. Reality

8.3.1 Hypothesis

Linear scaling would create **progressive reveal**:

- Early observations: small displacement (stay near true value)
- Later observations: larger displacement (fill the space)
- Golden ratio modulation provides optimal distribution

8.3.2 Reality (Empirical Results)

The algorithm produced **poor visual quality**:

- Excessive displacement for later observations
- Artificial separation misrepresents data structure
- Linear scaling inappropriate for this application

8.3.3 The Problem

For a tie group with $n = 100$ observations:

- Observation #1: displacement = 0% of ϵ
- Observation #50: displacement 49% of ϵ
- Observation #100: displacement 99% of ϵ

This creates a false impression of substructure where none exists.

8.4 Value of Negative Results

This negative result provides valuable scientific insights:

1. **Design pattern to avoid**: Linear scaling in displacement magnitudes
2. **Theoretical lesson**: Not all golden ratio applications succeed
3. **Empirical evidence**: Visual quality depends critically on scaling function
4. **Methodological contribution**: Systematic comparison reveals failures

9 Comparative Analysis

9.1 Dimensional Analysis

9.1.1 Three Distinct Approaches

Halton (Pure 1D):

- Operates entirely on a line
- Direct generation of 1D points
- No projection or transformation

Sunflower (2D $\rightarrow$ 1D):

- Constructs 2D polar coordinates (r, θ)
- Projects via $x = r\cos(\theta)$
- Preserves some 2D distribution properties

Intelligent (Hybrid 1D):

- Uses 1D displacement with 2D-inspired modulation
- Discards sine component, keeps cosine
- May combine disadvantages rather than advantages

9.2 Scaling Behavior Analysis

Method	Scaling Function	Growth Rate	At $n = 50$
Halton	Uniform $[-\epsilon/2, \epsilon/2]$	Constant	All uniform
Sunflower	$\propto \sqrt{i/n}$	Sublinear	$\sqrt{1}, \sqrt{2}, \sqrt{3}...$
Intelligent	$\propto i/n$	Linear	$1, 2, 3, 4, 5...$

Consequence: Intelligent creates artificial stratification that misrepresents uniform density.

9.3 Distribution Quality Metrics

9.3.1 Minimum Separation Distance

Measures the smallest gap between any two points:

- **Halton**: Guaranteed to decrease as $O(1/n)$
- **Sunflower**: Approximately $O(1/\sqrt{n})$ in projection
- **Intelligent**: Highly variable, depends on position

9.3.2 Visual Clustering

Qualitative assessment of artificial cluster formation:

- **Halton**: Minimal clustering, uniform coverage
- **Sunflower**: Slight central concentration, natural appearance
- **Intelligent**: Excessive separation, unnatural stratification

9.3.3 Perceptual Faithfulness

How well visual density matches actual data frequency:

- **Halton**: Excellent - uniform distribution reflects uniform ties
- **Sunflower**: Very good - slight radial gradient acceptable
- **Intelligent**: Poor - progressive displacement misleading

9.4 Computational Complexity

All three algorithms share similar computational profiles:

Operation	Complexity	Notes
Identify Ties	$O(n \log n)$	Sorting required
Generate Displacements	$O(k)$	Per tie group size k
Apply to Data	$O(n)$	Linear in observations
Overall	$O(n \log n)$	Dominated by tie detection

Practical Performance:

- Tested on datasets up to 100,000 observations
- Real-time interaction maintained (<100ms response)
- Memory-efficient implementation
- No performance degradation with large tie groups

10 User Study Design

10.1 Overview

Study Type: Within-subjects repeated measures design with counterbalancing

Target Sample: 30-50 participants

Participant Requirements:

- Basic visualization literacy
- No PCP expertise required
- Normal or corrected-to-normal vision
- No color blindness

10.2 Independent Variable

Jittering Method (5 levels):

1. No Jitter (baseline control)
2. Random Jitter (current practice benchmark)
3. Halton Jitter (low-discrepancy method)
4. Sunflower Jitter (biomimetic method)
5. Intelligent Jitter (novel method)

10.3 Dependent Variables

10.3.1 1. Task Accuracy (Primary)

Density Estimation Tasks:

- "How many observations follow this highlighted path?"
- "Which species is most common at this value?"
- Measured as absolute error from true count

Cluster Identification Tasks:

- "Identify the largest cluster in this group"
- "How many distinct sub-clusters exist?"
- Measured as classification accuracy

Pattern Detection Tasks:

- "Are there outliers in this tied group?"
- "Do these observations form a single cluster or multiple?"
- Measured as detection sensitivity and specificity

10.3.2 2. Task Completion Time (Secondary)

- Time to complete density estimation (seconds)
- Time to identify specific observations (seconds)
- Time to detect patterns (seconds)

10.3.3 3. Subjective Preference (Tertiary)

Quantitative Ratings:

- 10-point Likert scale for clarity
- 10-point Likert scale for ease of interpretation
- 10-point Likert scale for confidence in answers

Qualitative Feedback:

- Open-ended comments on clarity
- Perceived advantages/disadvantages
- Suggestions for improvement

Comparative Ranking:

- Forced ranking of all five methods
- Pairwise comparisons for fine-grained preferences

10.4 Hypotheses

H1 (Accuracy): Halton and Sunflower > Random > No Jitter

- Rationale: Uniform distribution enables accurate density perception

H2 (Time): Halton Sunflower < Random < No Jitter

- Rationale: Clear separation enables faster analysis

H3 (Preference): Sunflower Halton > Random > No Jitter

- Rationale: Aesthetic appeal combined with functionality

H4 (Intelligent): Performance below expectations across all metrics

- Rationale: Artificial patterns interfere with perception and interpretation

10.5 Statistical Analysis Plan

Primary Analysis:

- Repeated-measures ANOVA for each dependent variable
- Mauchly's test for sphericity
- Greenhouse-Geisser correction if sphericity violated

Post-hoc Comparisons:

- Bonferroni-corrected pairwise comparisons
- Effect size calculations (partial ^{2})
- Confidence intervals for mean differences

Supplementary Analyses:

- Correlation between accuracy and confidence ratings
- Order effects from counterbalancing
- Individual differences analysis

11 Integration with ggpcp Package

11.1 Software Architecture

The ggpcp package (VanderPlas et al. 2023) implements a **grammar of graphics** approach through three modular components:

1. **Variable Selection** (`pcp_select`): Choose and order dimensions
2. **Axis Scaling** (`pcp_scale`): Normalize or transform scales
3. **Tie Resolution** (`pcp_arrange`): Handle overlapping values ← **NEW**

11.2 Implementation Example

```
# Basic usage with Sunflower jitter
iris_plot <- iris %>%
  pcp_select(1:4) %>%
  pcp_scale(method = "uniminmax") %>%
  pcp_arrange(method = "sunflower")

# Advanced usage with custom parameters
iris_plot <- iris %>%
  pcp_select(Sepal.Length, Sepal.Width, Petal.Length, Petal.Width) %>%
  pcp_scale(method = "uniminmax") %>%
  pcp_arrange(
    method = "halton",
    epsilon = 0.08,
    numeric_ties = TRUE
  )

# Comparison of methods
library(patchwork)
p1 <- iris %>% pcp_select(1:4) %>% pcp_arrange(method = "none")
p2 <- iris %>% pcp_select(1:4) %>% pcp_arrange(method = "halton")
p3 <- iris %>% pcp_select(1:4) %>% pcp_arrange(method = "sunflower")

p1 / p2 / p3
```

11.3 Available Methods

Method	Description	Best For
`"none"`	No jitter (baseline)	Identifying the problem
`"random"`	Standard random jitter	Comparison benchmark
`"halton"`	Quasi-random low-discrepancy	Uniform distributions
`"sunflower"`	Biomimetic golden angle	Aesthetic + performance
`"intelligent"`	Golden ratio linear	Research/comparison only

11.4 Automatic Tie Detection

The implementation automatically:

- Identifies numerical ties within machine precision tolerance
- Groups tied observations
- Applies selected jittering method
- Preserves original data in metadata

11.5 User Interface Design Principles

1. **Sensible Defaults**: Works well without parameter tuning
2. **Progressive Disclosure**: Advanced options available but not required
3. **Consistent API**: Follows tidyverse conventions
4. **Informative Feedback**: Warns about potential issues
5. **Performance**: Optimized for interactive use

12 Theoretical Contributions

12.1 1. Cross-Domain Algorithm Adaptation

Biomimetic Optimization: First application of Vogel's phyllotaxis model (Vogel 1979) to statistical visualization.

- Demonstrates successful knowledge transfer from botany to data science
- Shows evolutionary optimization principles apply to visualization
- Opens pathway for other biomimetic approaches

Number-Theoretic Methods: Application of Halton sequences (Halton 1960) from computational geometry to statistical graphics.

- Brings rigorous mathematical guarantees to visualization
- Connects low-discrepancy theory to perceptual quality

- Establishes formal framework for distribution quality

12.2 2. Comparative Framework

First systematic comparison of tie-breaking methods for PCPs:

- Establishes evaluation criteria (determinism, uniformity, perceptual validity, scalability)
- Provides quantitative metrics for distribution quality
- Offers guidance for future algorithm development

12.3 3. Negative Result Documentation

Scientific value of Intelligent jitter failure:

- Identifies problematic design pattern (linear scaling)
- Demonstrates importance of scaling function choice
- Provides cautionary tale for golden ratio applications
- Contributes to methodological knowledge

12.4 4. Design Principles

Articulation of core requirements:

- **Determinism**: Necessary for scientific reproducibility (Peng 2011)
- **Uniformity**: Required for faithful density representation
- **Perceptual Validity**: Balances accuracy and visibility
- **Scalability**: Enables real-world application

These principles extend beyond PCPs to general visualization design.

13 Practical Impact

13.1 Immediate Benefits for Researchers

Reliable Density Visualization:

- Trustworthy assessment of observation counts
- Accurate perception of relative frequencies
- Detection of outliers within tied groups

Reproducible Figures:

- Identical visualizations across sessions
- Verifiable results for peer review
- Consistency in publications

Enhanced Exploratory Analysis:

- Pattern detection in complex data
- Cluster identification
- Multivariate relationship exploration

13.2 Implementation Advantages

Simple Interface:

```
# Single function call with default
data %>% pcp_arrange(method = "sunflower")

# All the complexity handled internally
```

Customizable Parameters:

- Epsilon magnitude adjustment
- Method selection
- Tie tolerance specification

Integration with Existing Workflows:

- Works with tidyverse pipes
- Compatible with ggplot2 ecosystem
- Minimal learning curve

13.3 Long-term Impact

Complete GPCP Framework: Completes the vision of VanderPlas et al. (2023):

- Categorical ties: Solved (existing)
- Numerical ties: Solved (new contribution)
- Unified approach to tie resolution

Methodological Contribution: Establishes:

- Evaluation methodology for jittering algorithms
- Performance benchmarks

- Design patterns for distribution algorithms

Foundation for Extensions:

- Adaptive parameter selection
- 2D scatter plot applications
- Other visualization contexts with overplotting

14 Future Work

14.1 1. Adaptive Epsilon Selection

Current Limitation: User must specify epsilon parameter.

Proposed Solution: Automatic, context-aware epsilon selection based on:

14.1.1 Data-Driven Heuristics

- Tie group size distribution
- Axis range and scale
- Number of dimensions
- Display resolution

14.1.2 Perceptual Models

Just-noticeable-difference thresholds (Weber 1834):

$$\Delta I / I = k$$

where: - I = stimulus intensity - ΔI = minimum perceivable change - k = Weber constant (0.02 for line length)

14.1.3 Optimization Framework

$$\epsilon^* = \arg\min_{\epsilon} \left[\alpha \cdot \text{Occlusion}(\epsilon) + \beta \cdot \text{Distortion}(\epsilon) \right]$$

where: - Occlusion(ϵ) measures remaining overlap - Distortion(ϵ) measures deviation from true values - α, β are user-specified weights

14.2 2. 2D Generalization

Motivation: Scatter plots face similar overplotting issues (Cleveland and McGill 1984).

14.2.1 Current Practice

- Random jitter with transparency
- Hexagonal binning
- Density contours

14.2.2 Proposed Extension

2D Sunflower: Use full (r, θ) coordinates without projection

$$x_j = \epsilon_x \cdot \sqrt{\frac{j-1}{n}} \cdot \cos((j-1) \times 137.508°)$$

$$y_j = \epsilon_y \cdot \sqrt{\frac{j-1}{n}} \cdot \sin((j-1) \times 137.508°)$$

2D Halton: Base-2 for x-axis, base-3 for y-axis (well-studied in computer graphics)

14.2.3 Research Questions

- How does 2D deterministic jitter compare to current methods?
- What are optimal epsilon values for x and y dimensions?
- How does method performance vary with tie group geometry?

14.3 3. Performance Optimization

14.3.1 Algorithmic Improvements

- **Hash-based tie detection**: $O(n)$ average case instead of $O(n \log n)$
- **Parallel processing**: Independent tie groups processed concurrently
- **GPU acceleration**: Large datasets benefit from parallel computation

14.3.2 Implementation Efficiency

- Vectorized operations in R
- C++ backend for critical paths via Rcpp
- Memory-efficient data structures
- Lazy evaluation strategies

14.3.3 Scalability Targets

- **Current**: 100K observations tested and functional
- **Goal**: 10M+ observations with real-time interaction
- **Stretch**: Streaming data support for dynamic visualizations

14.4 4. Extended Evaluation

14.4.1 Domain-Specific Studies

Medical Data:

- Electronic health records
- Patient trajectory visualization
- Clinical decision support systems

Financial Data:

- High-frequency trading patterns
- Portfolio analysis
- Risk assessment visualization

Scientific Data:

- Genomics (gene expression profiles)
- Climate science (model ensembles)
- Physics (particle collision data)

14.4.2 Expert User Evaluation

- Experienced data analysts
- Domain scientists
- Professional data journalists
- Information visualization researchers

14.4.3 Longitudinal Studies

- Learning effects over time
- Integration into real workflows
- Adoption patterns
- Long-term satisfaction

14.4.4 Alternative Tasks

- Hypothesis generation
- Anomaly detection (Johansson et al. 2005)
- Model diagnostics
- Uncertainty visualization

15 Conclusions

15.1 Summary of Problem and Solution

The Problem: Numerical ties in parallel coordinate plots create visual occlusion that:

- Obscures density information
- Hides substructure
- Prevents accurate pattern detection
- Compromises exploratory data analysis

The Solution: Three deterministic jittering algorithms:

1. **Sunflower**: Biomimetic approach with evolutionary optimization
2. **Halton**: Quasi-random method with mathematical guarantees
3. **Intelligent**: Novel approach that reveals design pitfalls

15.2 Key Findings

15.2.1 Low-Discrepancy Methods Superior

Sunflower and Halton jitter outperform random jitter because:

- Uniform distribution = faithful density representation
- Mathematical guarantees translate to perceptual benefits
- Elimination of clustering artifacts
- Predictable, interpretable results

15.2.2 Determinism Essential

Reproducibility in scientific visualization (Peng 2011) requires:

- Identical input → identical output
- Verifiable results
- Consistency across sessions
- Trustworthy publications

15.2.3 Scaling Function Critical

Intelligent jitter demonstrates that:

- Linear scaling creates artificial stratification
- Excessive displacement distorts perception
- Scaling function choice is as important as distribution algorithm
- Not all golden ratio applications succeed

15.2.4 General Principle

Uniformity + Determinism together provide optimal tie resolution for PCPs.

15.3 Significance of Contributions

15.3.1 Theoretical

- Cross-domain algorithm adaptation
- Mathematical framework for distribution quality
- Negative result documentation
- Design principles articulation

15.3.2 Methodological

- Three novel algorithms
- Comparative evaluation framework
- Implementation in production software
- Reproducible research materials

15.3.3 Practical

- Solves long-standing problem
- Simple user interface
- Integration with existing tools
- Immediate utility for analysts

15.4 Broader Implications

The principles and methods developed here extend beyond parallel coordinates to:

- **Scatter plots**: 2D overplotting with similar characteristics
- **Time series**: Repeated measurements creating ties
- **Network visualization**: Node positioning with constraints
- **General principle**: Wherever stochastic jitter is used, low-discrepancy alternatives merit consideration

15.5 Final Thoughts

This research completes the Grammar of Generalized Parallel Coordinates vision by providing systematic solutions for both categorical and numerical ties. The integration of biomimetic optimization and number-theoretic methods demonstrates the value of cross-disciplinary approaches in visualization research. The negative result with Intelligent jitter reminds us that not all theoretical ideas succeed in practice, and documenting failures contributes to scientific knowledge.

The development of deterministic jittering methods represents a step toward more trustworthy, reproducible, and effective multivariate data visualization. As datasets continue to grow in size and complexity, such systematic solutions become increasingly essential for extracting meaningful insights from high-dimensional data.

16 Timeline to Dissertation Defense

Phase	Timeframe	Key Milestones
Comprehensive Exam	Fall 2025	Presentation & committee approval
Algorithm Refinement	Winter 2025 – Spring 2026	Adaptive epsilon, optimization
User Study	Spring – Summer 2026	IRB approval, data collection, analysis

Phase	Timeframe	Key Milestones
Dissertation Writing	May – June 2026	Integration of all components
Defense	July 2026	Committee review and final defense

Target Defense Date: July 2026

17 References

Cleveland, William S, and Robert McGill. 1984. "Graphical Methods for Data Presentation: Full Scale Breaks, Dot Charts, and Multibased Logging." *The American Statistician* 38 (4): 270–80. https://doi.org/10.1080/00031305.1984.10483223.

Dennig, Frederik L, Tom Polk, Zeyuan Lan, Michael Han, Kathrin Ballweg, Michael Merz, Tobias Schreck, Daniel A Keim, and Min Chen. 2021. "ParSetgnostics: Quality Metrics for Parallel Sets." *Computer Graphics Forum* 40 (3): 375–86. https://doi.org/10.1111/cgf.14314.

Halton, John H. 1960. "On the Efficiency of Certain Quasi-Random Sequences of Points in Evaluating Multi-Dimensional Integrals." *Numerische Mathematik* 2 (1): 84–90. https://doi.org/10.1007/BF01386213.

Inselberg, Alfred. 1985. "The Plane with Parallel Coordinates." *The Visual Computer* 1 (2): 69–91. https://doi.org/10.1007/BF01898350.

Johansson, Jimmy, and Camilla Forsell. 2016. "Evaluation of Parallel Coordinates: Overview, Categorization and Guidelines for Future Research." *IEEE Transactions on Visualization and Computer Graphics* 22 (1): 579–88. https://doi.org/10.1109/TVCG.2015.2466992.

Johansson, Jimmy, Patric Ljung, Mikael Jern, and Matthew Cooper. 2005. "Revealing Structure Within Clustered Parallel Coordinates Displays." In *IEEE Symposium on Information Visualization*, 125–32. https://doi.org/10.1109/INFVIS.2005.1532141.

Kosara, Robert, Fabian Bendix, and Helwig Hauser. 2006. "Parallel Sets: Interactive Exploration and Visual Analysis of Categorical Data." *IEEE Transactions on Visualization and Computer Graphics* 12 (4): 558–68. https://doi.org/10.1109/TVCG.2006.76.

Niederreiter, Harald. 1992. *Random Number Generation and Quasi-Monte Carlo Methods*. Philadelphia, PA: Society for Industrial; Applied Mathematics. https://doi.org/10.1137/1.9781611970081.

Peng, Roger D. 2011. "Reproducible Research in Computational Science." *Science* 334 (6060): 1226–27. https://doi.org/10.1126/science.1213847.

VanderPlas, Susan, Yawei Ge, Antony Unwin, and Heike Hofmann. 2023. "Penguins Go Parallel: A Grammar of Graphics Framework for Generalized Parallel Coordinate Plots." *Journal of Computational and Graphical Statistics* 32 (4): 1405–20. https://doi.org/10.1080/10618600.2023.2181762.

Vogel, Helmut. 1979. "A Better Way to Construct the Sunflower Head." *Mathematical Biosciences* 44 (3-4): 179–89. https://doi.org/10.1016/0025-5564(79)90080-4.

Ware, Colin. 2012. *Information Visualization: Perception for Design*. 3rd ed. Waltham, MA: Morgan Kaufmann.

Weber, Ernst Heinrich. 1834. *De Pulsu, Resorptione, Auditu Et Tactu: Annotationes Anatomicae Et Physiologicae*. Leipzig: Koehler.

Wegman, Edward J. 1990. "Hyperdimensional Data Analysis Using Parallel Coordinates." *Journal of the American Statistical Association* 85 (411): 664–75. https://doi.org/10.1080/01621459.1990.10474926.

Wegman, Edward J, and Qiang Luo. 1991. "High Dimensional Clustering Using Parallel Coordinates and the Grand Tour." In *Computing Science and Statistics*, 28:352–60.

18 Appendix A: Algorithm Pseudocode

18.1 Sunflower Jitter

```
function sunflower_jitter(tied_values, epsilon):
    n = length(tied_values)
    golden_angle = 137.508 * (pi / 180)

    displacements = empty_array(n)

    for j from 1 to n:
        angle = (j - 1) * golden_angle
        radius = epsilon * sqrt((j - 1) / n)
        displacements[j] = radius * cos(angle)

    return tied_values + displacements
```

18.2 Halton Jitter

```
function van_der_corput(i, base):
    result = 0
    denominator = base
    while i > 0:
        result += (i mod base) / denominator
        i = floor(i / base)
        denominator *= base
    return result
```

```
function halton_jitter(tied_values, epsilon):
    n = length(tied_values)
    displacements = empty_array(n)

    for i from 0 to n-1:
        h = van_der_corput(i, 2)
        displacements[i+1] = epsilon * (h - 0.5)

    return tied_values + displacements
```

18.3 Intelligent Jitter

```
function intelligent_jitter(tied_values, epsilon):
    n = length(tied_values)
    golden_ratio = 0.618

    displacements = empty_array(n)

    for j from 1 to n:
        angle = (j - 1) * 2 * pi * golden_ratio
        scale = (j - 1) / n
        displacements[j] = epsilon * cos(angle) * scale

    return tied_values + displacements
```

19 Appendix B: Contact Information

Author: Denise Bradford
Email: denise.bradford@huskers.unl.edu
Institution: University of Nebraska–Lincoln, Department of Statistics
GitHub: https://github.com/drbradford12/Dissertation-Data

Dissertation Committee:

- [Committee members to be listed]

Document generated: October 2025
Last updated: [Date]