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

Slide 1: Title Slide

Speaking time: 30 seconds

Good morning/afternoon, committee members. Thank you for being here today. I'm Denise Bradford, and I'm presenting my comprehensive exam on resolving numerical ties in parallel coordinate plots. This work represents a critical advancement in high-dimensional data visualization, addressing a fundamental limitation that has plagued parallel coordinate plots since their inception. Today, I'll demonstrate how deterministic, low-discrepancy methods can transform cluttered, ambiguous visualizations into clear, interpretable displays.

Slide 2: Introduction

Speaking time: 15 seconds

Let me begin by providing context for this research and why addressing numerical ties is crucial for effective data visualization.

Slide 3: Motivation

Speaking time: 20 seconds

The motivation for this work comes directly from real-world challenges in data visualization, particularly from my experience developing interactive dashboards for data-driven decision making.

Slide 4: Chapter 1 - Rural Shrink Smart Dashboard

Speaking time: 2 minutes

This research originated from a very practical problem. In 2023, Dr. VanderPlas and I developed the Rural Shrink Smart dashboard - an interactive tool designed to help small Iowa towns understand and adapt to population decline. The dashboard visualized data for over 900 towns across more than 50 variables using parallel coordinate plots.

However, we encountered a critical issue: the dashboard was essentially unusable. Why? Because census data contains numerous tied values - multiple towns share the same median income, the same population density, the same educational attainment levels. These numerical ties created massive overplotting in the parallel coordinate visualization.

[Point to visualization] As you can see here, when multiple observations share identical values, they overlap perfectly on the axes. This visual clutter completely obscured individual community profiles. Town leaders couldn't compare their community to others - which was the dashboard's primary purpose. Without the ability to trace individual towns through the visualization, data-driven decisions became impossible.

We applied a temporary fix using deterministic jittering, which made the dashboard functional. But this experience revealed a fundamental gap in visualization methodology: there was no principled framework for handling numerical ties in parallel coordinate plots. That gap is what my dissertation aims to fill.

Slide 5: Research Overview

Speaking time: 1.5 minutes

The central challenge is this: numerical ties in parallel coordinate plots create severe visual occlusion that fundamentally compromises data exploration.

[Point to left visualization] Here's what happens with categorical ties - the ggpcp package already handles these well through hierarchical sorting. You can see distinct bands forming, and individual observations remain traceable.

[Point to right visualization] But here's the problem with numerical ties. Multiple observations collapse onto single points, creating complete visual occlusion. You cannot distinguish whether you're looking at one observation or one thousand.

Our approach addresses this through four key principles:

- Scientific reproducibility - the same data always produces the same visualization
- Perceptual validity - displacements are large enough to see but small enough to maintain data integrity
- Computational efficiency - methods that scale to real-world datasets
- Theoretical rigor - grounded in established mathematical frameworks

The expected contribution is a complete Grammar of Graphics framework for tie resolution in high-dimensional visualization.

Slide 6: Extensions Beyond PCPs

Speaking time: 45 seconds

While I'm focusing on parallel coordinate plots, this work has broader implications. The general principle is: wherever random jitter is currently used in visualization, deterministic low-discrepancy alternatives should be considered.

This applies to 2D scatter plots with overplotting, time series with multiple overlapping series, network layouts with node positioning constraints, and heatmaps with discrete values. The methods I'm developing provide reproducible, mathematically-grounded alternatives to random jittering across all these visualization types.

Slide 7: Problem Statement

Speaking time: 1 minute

Let me formally define the problem. Parallel coordinate plots, introduced by Inselberg in 1985 and popularized by Wegman in 1990, visualize n-dimensional data using parallel axes.

The numerical tie problem manifests in four critical ways:

- Visual collision: multiple observations overlap perfectly, making density unknowable
- Information loss: you cannot distinguish 1 from 1,000 observations
- Structural occlusion: any substructure within tied groups remains completely hidden
- Tracing becomes impossible: you cannot follow individual observations through the plot

Without solving these issues, parallel coordinate plots fail at their fundamental purpose: enabling exploration of high-dimensional data.

Slide 8: Research Context

Speaking time: 45 seconds

This work builds on four decades of research. Inselberg introduced PCPs in 1985, Wegman showed their statistical applications in 1990. The overplotting problem has been recognized for years, with various partial solutions proposed.

Schonlau introduced hammock plots in 2003 as an alternative that aggregates tied observations. VanderPlas and colleagues developed the ggpcp framework in 2023, providing a Grammar of Graphics foundation and solving categorical ties.

My contribution completes this trajectory by adding numerical tie resolution to create a comprehensive solution.

Slide 9: The ggpcp Foundation

Speaking time: 1 minute

The ggpcp package provides an elegant three-module architecture that separates concerns:

Module 1, pcp_select, handles dimension selection and ordering - this is complete. Module 2, pcp_scale, normalizes scales across axes - also complete. Module 3, pcp_arrange, handles tie resolution - this is what I'm extending.

The existing implementation handles categorical ties beautifully through hierarchical sorting. My extension adds numerical tie handling through deterministic jittering with uniform constraints. The result is a complete tie-handling framework that maintains the Grammar of Graphics philosophy.

Slide 10: Terminology and Definitions

Speaking time: 15 seconds

Let me establish some key terminology that will be used throughout the presentation.

Slide 11: Core Concept - Numerical Ties

Speaking time: 1 minute

Numerical ties are multiple observations sharing identical numerical values, causing perfect visual overlap.

These arise from four main sources:

- Rounding: when data is recorded with limited precision - heights to the nearest tenth of a meter
- Instruments: discrete sensors that produce integer counts
- Natural clustering: when values naturally cluster, like Likert scale responses
- Encoding: when categorical data is converted to numeric form

The key distinction here is between categorical ties, which are expected and handled through sorting, versus numerical ties, which are data-driven and require displacement to resolve.

Slide 12: Core Concept - Adaptive Uniform Jittering

Speaking time: 1.5 minutes

When multiple observations create ties, we face several problems: they stack on identical positions, visual density becomes completely misleading, individual trajectories are impossible to follow, and pattern detection is severely compromised.

Our solution applies adaptive uniform jittering where the interval width epsilon equals 1 divided by the number of ties. This means the more observations that share a value, the smaller the displacement for each individual observation.

Key properties of this approach:

- The interval adapts to tie frequency - more ties mean finer resolution
- The expected value is preserved - the center of the jittered points remains at the original value
- No overlap occurs between distinct values - we maintain value separation
- The method scales elegantly for any number of ties

This ensures we reveal structure within tied groups while maintaining data integrity.

Slide 13: Background and Motivation

Speaking time: 15 seconds

Now let's examine the specific challenges and existing approaches in more detail.

Slide 14: The Problem - Visual Overlap

Speaking time: 1.5 minutes

[Point to visualization] This figure perfectly illustrates the severity of the problem. Look at the numerical ties in the ggpcp visualization - you see complete occlusion where multiple observations share values.

We face three critical issues:

First, visual collision - when observations overlap perfectly, we lose all information about count and density. You literally cannot tell if you're looking at one observation or hundreds.

Second, density information loss - this makes it impossible to identify which values are common versus rare, destroying a fundamental aspect of data exploration.

Third, structural occlusion - any patterns, sub-clusters, or relationships within the tied groups remain completely hidden.

The impact on analysis is severe. Blumenschein and colleagues showed in 2020 that cluster identification becomes compromised. Outlier detection becomes impossible when you can't see individual observations. And pattern tracing - the fundamental strength of parallel coordinates - is completely broken.

Slide 15: Existing Solutions - Categorical Ties

Speaking time: 1.5 minutes

The ggpcp package already provides an elegant solution for categorical ties through hierarchical sorting implemented in pcp_arrange.

[Point to visualization] Notice how categorical values are spread evenly across their range, with observations ordered to minimize line crossings. This creates these beautiful band patterns that are visually similar to parallel sets when the data is dense.

The key benefits are:

- Reduced line crossings make patterns clearer
- Individual observations remain traceable
- The computational ordering serves as "external cognition" - the computer does the work of untangling, reducing cognitive load on the viewer

This works wonderfully for categories because their discrete nature naturally supports hierarchical ordering and equispaced distribution. But this same approach cannot be directly applied to numerical data.

Slide 16: Alternative Approach - Hammock Plots

Speaking time: 2 minutes

Hammock plots, introduced by Schonlau in 2003, take a fundamentally different approach. Instead of showing individual observations, they aggregate tied values into parallelograms where the width is proportional to the number of observations.

[Point to UC Berkeley admissions visualization] Look at how hammock plots handle the same data. The width of each band directly encodes frequency. Multiple tied observations become wider boxes rather than overlapping lines.

This aggregation strategy has three key advantages:

- Explicit density representation through visual magnitude
- No occlusion even with vastly different frequencies
- Seamless handling of mixed categorical and numerical variables

However, there are significant limitations:

- You lose the ability to trace individual observations
- The visualization requires more white space, especially with frugal spacing
- Continuous variables often require binning, which loses precision

This trade-off - aggregate clarity versus individual traceability - is fundamental to the difference between hammock plots and our approach.

Slide 17: Comparison - Hammock vs GPCP

Speaking time: 1 minute

Let me highlight the key differences between these approaches:

Hammock plots use constant-width boxes between numerical variables, while ggpcp shows individual lines that overlap. For categorical to numerical transitions, hammock plots maintain constant width while ggpcp creates these characteristic triangular shapes.

Hammock plots excel when:

- You're emphasizing bivariate relationships between adjacent axes

- Your dataset has many observations per value
- You're focused on aggregate patterns rather than individual trajectories

But ggpcp with proper tie-breaking is superior when you need to trace individual observations or work with smaller datasets where individual cases matter.

Slide 18: Why ggpcp Needs a Different Solution

Speaking time: 1 minute

Despite hammock plots' success, ggpcp requires a different approach for numerical ties. Here's why:

First, preserving individual traceability is a core feature of ggpcp - users expect to follow observations through the high-dimensional space.

Second, the Grammar of Graphics philosophy naturally accommodates position adjustments as a transformation step.

Third, we want flexibility - users should be able to choose between aggregation and separation based on their analytical needs.

Fourth, for small to medium datasets, seeing individual observations provides crucial insights that aggregation would hide.

There's also an exciting integration opportunity for future work: we could combine jittering for position separation with width encoding for density, getting the benefits of both approaches.

Slide 19: Design Requirements

Speaking time: 20 seconds

Our solution must satisfy four core principles to be effective. Let me walk through each of these critical requirements.

Slide 20: Four Core Principles

Speaking time: 1.5 minutes

Principle 1: Determinism. Every run with the same data must produce identical results. This is non-negotiable for scientific reproducibility, as Roger Peng emphasized in his 2011 Science paper. We achieve this through algorithmic guarantees with no random number generation.

Principle 2: Uniformity. Points must be evenly distributed within the jittering interval. This ensures faithful density representation and minimizes visual artifacts that could suggest false patterns. We need systematic space-filling approaches, not random scattering.

Principle 3: Perceptual Validity. This requires a careful balance - displacement must be small enough to maintain data integrity and user trust, yet large enough for visual distinction. As Colin Ware's perception research shows, this balance is crucial for effective visualization.

Slide 21: Four Core Principles (continued)

Speaking time: 1 minute

Principle 4: Scalability. The method must handle everything from 2 to 1000+ observations efficiently. We need O(n) complexity for interactive performance and linear memory requirements for feasibility.

These principles aren't just nice-to-have features - they're essential for creating a visualization method that practitioners will actually use and trust.

Slide 22: Mathematical Framework

Speaking time: 20 seconds

Now let's examine the mathematical foundations that make this approach rigorous and principled.

Slide 23: Core Constraint (Uniform Version)

Speaking time: 1.5 minutes

For tied value v with n_ties observations, we distribute points uniformly within the interval from v minus 1 over 2n_ties to v plus 1 over 2n_ties.

This formulation is elegant in its simplicity but powerful in its implications. The interval width scales inversely with the number of tied observations - the more ties you have, the finer the resolution of separation.

The key properties are mathematically guaranteed:

- The interval adapts automatically to tie frequency
- The expected value remains exactly v - we preserve the data's central tendency
- Distinct values never overlap - we maintain clear value separation
- The method scales gracefully from 2 to thousands of ties

This adaptive behavior is crucial - it means heavily tied values get finer resolution while less common ties can use more space, optimizing the use of visual real estate.

Slide 24: Halton Jitter - Overview

Speaking time: 1 minute

Now we move beyond simple uniform distribution to Halton sequences - a sophisticated approach from number theory introduced by John Halton in 1960.

The problem with random jittering is clustering. With 100 random points, you expect gaps of 0.05 but might get some as small as 0.0001. The birthday paradox tells us that random points will cluster, creating visual artifacts.

Halton sequences solve this through systematic, deterministic gap-filling. Each new point is placed to optimally fill the largest remaining gap. This creates predictable, uniform coverage with mathematical guarantees on distribution quality.

Slide 25: Halton Jitter - Van der Corput Construction

Speaking time: 1.5 minutes

The Van der Corput sequence, which forms the basis of Halton sequences, uses an elegant bit-reversal algorithm.

Here's how it works: Take integer index i, convert it to binary, reverse the binary digits, then interpret the result as a fraction.

[Point to visualization] Look at the comparison between Halton and pseudorandom points. The Halton sequence on the left shows systematic, even coverage - each point bisects the largest gap. The pseudorandom points on the right show obvious clustering and gaps.

This isn't just visually better - it's mathematically optimal. Each point is placed to maximize the minimum distance to existing points, creating the most uniform distribution possible.

Slide 26: Halton Jitter - Theoretical Guarantees

Speaking time: 1 minute

The mathematical foundations come from discrepancy theory, formalized by Niederreiter in 1992. Star discrepancy measures how uniformly points fill space.

Random sequences achieve O(n^-0.5) discrepancy - relatively poor uniformity. Halton sequences achieve O(n^-1 log n) - near-optimal uniformity. The theoretical limit is Omega(n^-1 log n), meaning Halton sequences are essentially as good as mathematically possible.

These aren't just abstract mathematical properties - they translate directly to visual quality. Better discrepancy means more uniform point distribution, fewer visual artifacts, and more faithful representation of data density.

Slide 27: Empirical Evidence and Perceptual Foundations

Speaking time: 20 seconds

Let's examine the perceptual research that informs our design decisions.

Slide 28: Perceptual Challenges in Parallel Coordinates

Speaking time: 1.5 minutes

Hofmann's 2013 research revealed a crucial perceptual phenomenon: users don't perceive vertical distance between parallel lines - they perceive perpendicular distance.

[Point to top figure] When lines are at an angle, the perpendicular distance is less than the vertical distance. This means users systematically underestimate separation when lines aren't horizontal.

[Point to bottom figure] Day's 1991 sine illusion shows another challenge - equal-length vertical segments in a sine wave pattern appear unequal due to the surrounding context.

The implications for tie resolution are significant:

- Users judge separation based on orthogonal distance, not axis distance
- Our epsilon parameter must account for this perceptual bias
- Different line angles may require different epsilon values for perceptually equivalent separation

This is why empirical validation through user studies is essential - mathematical uniformity doesn't guarantee perceptual uniformity.

Slide 29: Clutter and Overplotting

Speaking time: 1.5 minutes

Johansson and Forsell's 2016 comprehensive review found that even medium-sized datasets suffer from severe overplotting in parallel coordinates, creating displays too cluttered to perceive trends or structure.

Ties exacerbate this problem exponentially. Without ties, you get overplotting from values in similar ranges. With ties, you get perfect overlap with complete occlusion and no frequency information whatsoever.

Existing clutter reduction approaches each have limitations: - Clustering methods using bands or envelopes lose individual tracing ability - Transparency and density techniques fail completely with perfect overlap - alpha blending doesn't help when lines are exactly coincident

Our uniformity adjustment approach is different - we resolve ties before rendering, preserving individual traces while revealing density. This complements rather than replaces other clutter reduction methods.

Slide 30: Dimension Ordering Effects

Speaking time: 1 minute

Peng, Ward, and Rundensteiner's 2004 work showed that dimension ordering is crucial for PCP effectiveness. It's an NP-complete problem requiring heuristic solutions.

This interacts with tie resolution in important ways. Different dimension orderings reveal different tie patterns. Ties between dimensions A and B might be highly visible in one ordering but hidden in another.

This means our tie detection must be axis-pair specific, evaluation must consider multiple orderings, and integration with ggpcp's existing pcp_select function maintains the flexibility users need.

The key insight: tie resolution and dimension ordering are interdependent problems that must be solved together.

Slide 31: Cluster Identification Performance

Speaking time: 1 minute

Recent studies by Holten and van Wijk in 2010 and Blumenschein in 2020 evaluated cluster identification in PCPs. They found that optimal configurations depend heavily on task type and cluster characteristics.

The critical question for us: Does tie-breaking help or hinder cluster identification?

Potential benefits include revealing hidden clusters within tied groups, improving cluster boundary visibility, and enabling accurate cluster size estimation.

But there are risks: displacement might obscure naturally tight clusters, added visual complexity increases cognitive load, and poor jittering could create artifacts suggesting false clusters.

This is why our user study will specifically test cluster identification tasks.

Slide 32: Task-Dependent Performance

Speaking time: 1 minute

Johansson and Forsell's 2016 work showed PCP effectiveness varies significantly by task complexity.

For density estimation - a simple task - we expect large improvements with clear separation.

For cluster identification - medium complexity - tie-breaking should reveal structure but sensitivity to method choice matters.

For outlier detection - complex task - tie resolution is essential for individual line tracing.

Our evaluation will use these validated task categories from the literature, controlling for confounds like ordering and design, measuring multiple outcomes including accuracy, time, and confidence, and comparing against baselines.

Slide 33: Practical Design Recommendations

Speaking time: 45 seconds

Current best practices for parallel coordinates include five principles: manage visual clutter, optimize dimension ordering, consider perceptual factors, support interaction, and use appropriate encodings.

We're adding a crucial sixth principle: Resolve numerical ties uniformly. Apply jittering to prevent perfect overlap, use low-discrepancy methods like Halton sequences, integrate with existing capabilities, and maintain reproducibility.

This isn't just an add-on - it's a fundamental requirement for effective parallel coordinate visualization that integrates with all other principles.

Slide 34: Chapter 2 - Integration with ggpcp Package

Speaking time: 20 seconds

Now let me show you how these theoretical concepts translate into practical implementation within the ggpcp framework.

Slide 35: Comparison - Uniform vs Halton

Speaking time: 1 minute

[Point to visualizations] Here's a direct comparison of jittering methods on simple data.

The original shows perfect overlap - no information about density. Uniform jittering spreads points randomly within intervals - functional but irregular. Halton sequences create systematic, visually pleasing distributions with optimal spacing.

The Halton approach isn't just aesthetically superior - it provides consistent, predictable results that users can trust.

Slide 36: Method Characteristics

Speaking time: 45 seconds

Let me summarize the key characteristics of each method:

Uniform jittering offers simple implementation and preserves statistical properties, but may create irregular spacing that can be visually distracting.

Halton sequences provide quasi-random distribution with regular spacing and excellent visual quality, though with slightly higher computational complexity - O(n log n) versus O(n).

For exploratory analysis where speed matters most, uniform jittering works well. For publication graphics or presentations where visual quality is paramount, Halton sequences are worth the extra computation.

Slide 37: Integration with PCPs

Speaking time: 1 minute

[Point to before/after] Here's the transformation in action. The original plot shows severe overplotting - you cannot distinguish individual paths or assess density.

After applying jittering, individual paths become visible. You can now trace observations through the plot, identify patterns, and understand the true data distribution.

This isn't just a cosmetic improvement - it fundamentally changes what insights are possible from the visualization.

Slide 38: Pseudocode Implementation

Speaking time: 1.5 minutes

[Point to three panels] This comparison shows the complete framework in action on the iris dataset.

The left panel shows the original data with severe numerical ties. You can see complete occlusion in the zoomed regions.

The middle panel applies uniform jittering with epsilon equal to 1/n. Points are separated but with irregular spacing that could suggest false patterns.

The right panel uses Halton sequence jittering. Notice the systematic, regular spacing that maintains visual clarity while accurately representing density.

This demonstrates why method choice matters - same data, same epsilon value, but very different visual results and potential for misinterpretation.

Slide 39: Implementation Guidelines

Speaking time: 1 minute

When should you apply jittering? Four key scenarios:

- Categorical variables in parallel coordinates always need it
- Discrete numeric data with many ties benefits greatly
- Mixed data types require careful handling of both categorical and numerical ties
- Large datasets where overplotting obscures patterns

For method choice: use uniform jittering for exploratory analysis where speed matters, Halton sequences for publication graphics where quality is paramount, and consider user-controlled epsilon for interactive visualizations.

The integration with ggpcp is seamless - just a few lines of code to transform an unusable visualization into an insightful one.

Slide 40: Architectural Integration

Speaking time: 45 seconds

The integration respects ggpcp's elegant three-module architecture. We're extending only the third module - pcp_arrange - to handle numerical ties alongside the existing categorical tie-breaking.

This design maintains backward compatibility, follows Grammar of Graphics principles, and integrates seamlessly with existing workflows. Users don't need to learn a new system - just new options within the familiar framework.

Slide 41: Proposed Implementation

Speaking time: 1 minute

Here's the proposed function signature. We're adding three new parameters to pcp_arrange:

The method parameter now includes "halton" alongside the existing "from-left" and "from-right" options.

Epsilon controls maximum displacement - NULL for automatic determination at 5% of axis range, or a user-specified numeric value for precise control.

The numeric_ties boolean enables or disables numerical jittering, defaulting to TRUE for convenience.

This API design prioritizes ease of use while providing fine-grained control when needed.

Slides 42-44: Example Usage

Speaking time: 1.5 minutes (total for examples)

[Slide 42] Here's basic usage with the iris dataset. Just a few lines of code: select your dimensions, apply Halton arrangement with specified epsilon, then plot. The result is immediately more interpretable.

[Slide 43] For mixed data types, the method handles both categorical and numerical ties simultaneously. Halton jittering applies to numerical variables while the space parameter controls categorical spacing.

The comparison example shows how easy it is to evaluate different methods side-by-side using patchwork for layout.

[Slide 44] Documentation will include comprehensive function documentation, theoretical foundations, parameter selection guidance, and multiple vignettes covering everything from basic usage to advanced techniques.

Slide 45: Research Questions and Methodology

Speaking time: 20 seconds

Now let's examine the specific research questions that will guide the remainder of this work.

Slide 46: Primary Research Question

Speaking time: 45 seconds

The overarching question is: How can the formal structure of the Grammar of Graphics be extended to systematically incorporate and evaluate methods for resolving numerical ties in parallel coordinate plots, and what is the quantifiable impact of these methods on the accuracy and efficiency of visual data analysis?

This question bridges theory and practice, requiring both formal mathematical development and empirical validation.

Slides 47-51: Research Questions 1-4

Speaking time: 4 minutes (1 minute per RQ)

[RQ1 - Slide 47] The first question addresses theory: How can tie management be formalized within the Grammar of Graphics? We'll use theoretical analysis, literature synthesis, and formal specification to extend the grammar notation and create integration guidelines for ggpcp.

[RQ2 - Slide 48] The second question tackles methodology: What are the optimal algorithmic criteria for ordering and spacing tied points? We'll design algorithms, analyze distribution quality, benchmark performance, and identify edge cases. Success metrics include minimum separation distance, uniformity measures, and computational efficiency.

[RQ3 - Slides 49-50] The third question examines perception through a within-subjects study with 100-150 analysts. Participants will complete four tasks - density estimation, cluster identification, outlier detection, and pattern tracing - using different methods. We'll measure accuracy, time, confidence, and preference.

Our hypotheses predict Halton will show higher accuracy but potentially longer completion times than no jitter. We expect good but not perfect performance - artificial patterns might interfere with perception.

[RQ4 - Slide 51] The fourth question develops practical guidance. We'll synthesize findings into evidence-based heuristics, create decision trees, validate with case studies across six domains from bioinformatics to climate science, and refine through practitioner feedback.

Slides 52-57: Implementation Roadmap

Speaking time: 3 minutes (30 seconds per phase)

[Slide 53] Phase 1 this winter focuses on algorithm refinement - finalizing implementations, developing adaptive epsilon selection, optimizing performance, and comprehensive testing.

[Slide 54] Phase 2 in spring extends ggpcp - implementing automatic tie detection, epsilon auto-determination, visual indicators, and preparing for CRAN submission.

[Slide 55] Phase 3 in spring/summer conducts the user study - IRB approval, participant recruitment, data collection, and statistical analysis leading to an empirical paper.

[Slide 56] Phase 4 in summer applies methods to real datasets, gathers practitioner feedback, develops decision heuristics, and writes dissertation chapters.

[Slide 57] Phase 5 in May-July 2026 covers final review, incorporating committee feedback, and the dissertation defense.

Slides 58-62: Expected Outcomes

Speaking time: 2 minutes (30 seconds per outcome type)

[Slide 59] Theoretical contributions include extending the Grammar of Graphics with biomimetic transformations, demonstrating cross-domain algorithm adaptation, and documenting negative results as learning opportunities.

[Slide 60] Methodological contributions provide novel algorithms, systematic evaluation frameworks, and production-ready implementations with comprehensive testing.

[Slide 61] Practical contributions enhance the ggpcp package with complete tie-handling, evidence-based guidance, and real-world impact on exploratory data analysis.

[Slide 62] Empirical contributions include quantitative performance data, diverse case studies, and benchmark datasets for future research.

Slides 63-65: Broader Implications

Speaking time: 1.5 minutes

[Slide 64] These methods generalize beyond parallel coordinates. 2D scatter plots can use full 2D Halton for overplotting. Time series can apply vertical displacement for overlapping series. Network visualizations can optimize node positioning with uniform principles.

[Slide 65] The general principle is profound: wherever random jitter is currently used, deterministic low-discrepancy alternatives should be considered. This provides reproducibility, better distribution quality, elimination of artifacts, and theoretical guarantees.

This work establishes design patterns applicable across visualization domains and principles for perceptually-valid position adjustments.

Slide 67: Timeline

Speaking time: 45 seconds

The timeline is aggressive but achievable. Algorithm refinement through winter 2025, ggpcp integration in spring, user study in spring/summer, case studies and writing through summer, final revisions in May-June 2026, and defense in July 2026.

Each phase builds on the previous, with built-in flexibility for iteration based on findings.

Slides 69-70: Conclusion and Key Findings

Speaking time: 2 minutes

To summarize: Numerical ties create severe visual occlusion in parallel coordinate plots, preventing density visualization, observation tracing, cluster identification, and pattern detection.

Our solution uses deterministic methods - particularly Halton sequences with mathematical guarantees - to systematically resolve these ties.

The impact is a complete framework extending the Grammar of Graphics, enabling reproducible, high-quality visualizations that reveal previously hidden patterns.

Key findings from our preliminary work:

- Low-discrepancy methods are definitively superior to random approaches
- Determinism is essential for scientific reproducibility
- Uniform distribution translates to faithful density representation
- Mathematical guarantees produce perceptual benefits

We've also learned important lessons about what doesn't work - linear scaling creates excessive displacement that distorts perception. The scaling function is as critical as the distribution algorithm.

Slide 71: Questions for Discussion

Speaking time: 1 minute

I'd like to pose several questions for committee discussion:

Should adaptive epsilon be user-overrideable or always automatic? There are arguments for both approaches.

Should the package default to Halton given its superiority, or maintain backward compatibility?

Are there additional task types or datasets you'd recommend for the user study?

What's the optimal publication strategy - one comprehensive paper or multiple focused papers targeting different communities?

Slide 72: Acknowledgments

Speaking time: 30 seconds

Thank you to my dissertation committee for your invaluable guidance and feedback, the ggpcp package developers for creating the foundation for this work, the UNL Statistics Department for support, pilot study participants who helped refine methods, and the broader R community for fostering open scientific software.

I welcome your questions and look forward to our discussion.

Slides 73-75: References

Speaking time: 10 seconds

The complete references are provided for your review. I'm happy to discuss any of these papers in more detail during our question period.

Additional Notes for Q&A Preparation

Anticipated Questions and Detailed Responses

Q1: "Why not just use random jittering? It's simpler and well-understood."

Response: Random jittering has three fundamental problems:

1. **Reproducibility**: Scientific visualization requires reproducible results. Random jittering produces different outputs each run, making it impossible to replicate findings or create consistent publications.
2. **Distribution quality**: Random points cluster due to the birthday paradox. With 100 points, you expect uneven gaps and clusters that can suggest false patterns.
3. **No guarantees**: Random methods provide no theoretical guarantees on uniformity or minimum separation distance.

Halton sequences solve all three issues while maintaining similar computational complexity.

Q2: "How does this relate to existing work on clutter reduction?"

Response: Our work is complementary to, not competitive with, existing clutter reduction techniques. Methods like alpha blending, density estimation, and edge bundling address different aspects of the overplotting problem.

Specifically:

- Alpha blending helps when lines cross at different points but fails with perfect overlap
- Density estimation aggregates information but loses individual traces
- Edge bundling reduces visual complexity but doesn't address ties

Our method resolves ties at the data transformation level, before rendering. This means it can be combined with any of these other techniques for even better results. Think of it as fixing the data representation problem rather than just the rendering problem.

Q3: “What about computational performance for large datasets?”

Response: Performance is a key design consideration:

- Uniform jittering is O(n) - linear in the number of observations
- Halton sequences are O(n log n) due to the bit reversal operations
- Both have linear memory requirements

For a dataset with 10,000 observations:

- Uniform jittering: ~10ms
- Halton sequences: ~150ms

This difference is negligible for interactive visualization. For massive datasets (100,000+ observations), we could offer adaptive strategies - Halton for visible regions, uniform for overview.

Q4: “How do you validate that your perceptual assumptions are correct?”

Response: This is exactly why RQ3 includes a comprehensive user study. We’re testing four different tasks with mixed expertise levels to ensure our mathematical optimality translates to perceptual effectiveness.

Additionally, we’re building on established perceptual research:

- Hofmann’s work on distance perception in parallel coordinates
- Ware’s principles of information visualization
- Johansson and Forsell’s task-based evaluation framework

The study will measure not just accuracy but also confidence and preference, ensuring the methods are both effective and trustworthy.

Q5: “What if users want to see the original, untransformed data?”

Response: Great question - transparency and user control are essential. The implementation includes:

- A boolean flag to disable jittering entirely
- Visual indicators showing where jittering is applied
- Optional overlay of original positions
- Complete documentation of transformations in metadata

Users always have full control and awareness of any transformations applied to their data.

Q6: “How does this handle mixed continuous and discrete numerical data?”

Response: The method adapts automatically to data characteristics:

- Pure continuous data with no ties: no jittering applied
- Discrete data with ties: automatic tie detection and resolution
- Mixed data: ties are resolved while maintaining relative positions of non-tied values

The epsilon parameter can be adjusted per axis if different scales require different handling. The key is that the method is adaptive - it only intervenes where necessary.

Q7: "What about the curse of dimensionality? Does this scale to 50+ dimensions?"

Response: Parallel coordinates themselves are one of the few techniques that handle high dimensionality well. Our tie resolution actually improves scalability by:

- Making patterns visible that would otherwise be hidden
- Reducing the cognitive load of interpreting overlapped data
- Enabling dimension reduction techniques to work more effectively

The computational complexity remains linear in the number of dimensions, so there's no algorithmic barrier to scaling.

Q8: "How does this relate to Grand Tour and other dynamic visualization methods?"

Response: Excellent connection! Dynamic methods like Grand Tour provide different projections over time, while we ensure each static projection is maximally informative.

The methods are highly complementary:

- Tie resolution ensures each frame of a Grand Tour is interpretable
- Dynamic transitions help verify that patterns revealed by tie-breaking are real
- Combined approach provides both static clarity and dynamic exploration

Future work could explore optimizing tie resolution for smooth transitions in animated visualizations.

Q9: "What about alternative approaches like dimensionality reduction before visualization?"

Response: Dimensionality reduction (PCA, t-SNE, UMAP) and tie resolution solve different problems:

- Dimensionality reduction loses information about original variables
- Tie resolution preserves all original dimensions and relationships

They're complementary strategies: 1. Apply tie resolution to see all patterns in original space 2. Use insights to inform dimensionality reduction choices 3. Apply tie resolution to the reduced space if needed

Our method ensures you can see what you have before deciding what to keep.

Q10: "How do you know the 'optimal' epsilon value?"

Response: This is an active area of research. Currently, we use:

- Default: 5% of axis range (empirically effective)
- Adaptive: inversely proportional to tie frequency
- User override: for domain-specific requirements

The user study will help establish perceptually optimal ranges. We're also developing interactive tools for epsilon tuning with real-time visual feedback.

Technical Deep Dives (if asked for more detail)

Van der Corput Sequence Mathematics

The base-b Van der Corput sequence for integer i:

1. Express i in base b: $i = \Sigma(a_k * b^k)$
2. Reverse the digits: $\Sigma(a_k * b^(-k-1))$
3. Result is in [0,1] with optimal discrepancy

Discrepancy Theory

Star discrepancy D*_n measures the maximum deviation between actual and expected proportion of points in any axis-aligned box:

- Random: $E[D*_n] = O(n^(-1/2))$
- Halton: $D*_n = O(n^(-1) * (logn)^d)$
- Optimal: $\Omega(n^(-1) * (logn)^(d-1))$

Grammar of Graphics Extension

Traditional: Statistical Transformation → Geometric Objects → Position Scales → Coordinate System
Extended: Statistical Transformation → Geometric Objects → Position Scales → **Position Adjustment** → Coordinate System

The position adjustment stage handles both categorical arrangement and numerical tie resolution.

Broader Impact Statement

This work advances the field of data visualization by:

1. Making high-dimensional data exploration more accessible to non-experts
2. Enabling discoveries in datasets previously too cluttered to analyze
3. Establishing principles applicable across visualization domains
4. Contributing to reproducible science through deterministic methods
5. Bridging mathematical theory and practical visualization

The immediate beneficiaries include:

- Data scientists exploring complex datasets
- Domain experts without visualization training

- Researchers requiring reproducible graphics
- Software developers building visualization tools

The long-term impact extends to any field requiring high-dimensional data analysis: genomics, finance, climate science, social science, and beyond.