Simplicity in Relational Structures

R.L.F. Brignall

School of Mathematics and Statistics
University of St Andrews

Wednesday 6th June, 2007
Introduction
A relational structure: a set of points, and a set of relations on these points.

- The ground set, $A$.
- A $k$-ary relation $R$ – a subset of $A^k$. 
A relational structure: a set of points, and a set of relations on these points.

The ground set, $A$.

A $k$-ary relation $R$ – a subset of $A^k$. 
A relational structure: a set of points, and a set of relations on these points.

The ground set, \( A \).

A \( k \)-ary relation \( R \) – a subset of \( A^k \).
Permutations

- A permutation of length $n$ is a structure on two linear relations.

Example

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
A permutation of length $n$ is a structure on two linear relations.

Example:

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
A permutation of length $n$ is a structure on two linear relations.

Example

$1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.$

8
A permutation of length $n$ is a structure on two linear relations.

Example

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
8 ⪯ 5
A permutation of length $n$ is a structure on two linear relations.

Example

1 \(<\) 2 \(<\) 3 \(<\) 4 \(<\) 5 \(<\) 6 \(<\) 7 \(<\) 8 \(<\) 9.

8 \(<\) 5 \(<\) 2
A permutation of length $n$ is a structure on two linear relations.

**Example**

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
8 ≺ 5 ≺ 2 ≺ 3
A permutation of length $n$ is a structure on two linear relations.

Example

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
8 < 5 < 2 < 3 < 9
A permutation of length $n$ is a structure on two linear relations.

Example

$1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9$.

$8 \prec 5 \prec 2 \prec 3 \prec 9 \prec 6$
A permutation of length $n$ is a structure on two linear relations.

Example

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.

8 < 5 < 2 < 3 < 9 < 6 < 7
A permutation of length $n$ is a structure on two linear relations.

Example

1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.
8 < 5 < 2 < 3 < 9 < 6 < 7 < 4
A permutation of length $n$ is a structure on two linear relations.

**Example**

- $1 < 2 < 3 < 4 < 5 < 6 < 7 < 8 < 9.$
- $8 < 5 < 2 < 3 < 9 < 6 < 7 < 4 < 1$
A graph is a relational structure on a single binary symmetric relation.
A graph is a relational structure on a single binary symmetric relation.
A **poset** is a relational structure on a binary reflexive antisymmetric transitive relation.
A poset is a relational structure on a binary reflexive antisymmetric transitive relation.
Tournaments

- A **tournament** is a complete oriented graph.
- As a relational structure, it is a single trichotomous binary relation. \((x \rightarrow y, y \rightarrow x \text{ or } x = y.\)
- A **competition** between players: \(x \rightarrow y\) means “y wins.”

Example
Tournaments

- A **tournament** is a complete oriented graph.
- As a relational structure, it is a single trichotomous binary relation. \((x \rightarrow y, y \rightarrow x \text{ or } x = y)\)
- A **competition** between players: \(x \rightarrow y\) means “y wins.”

Example
A tournament is a complete oriented graph.

As a relational structure, it is a single trichotomous binary relation. ($x \rightarrow y$, $y \rightarrow x$ or $x = y$.)

A competition between players: $x \rightarrow y$ means “$y$ wins.”
Tournaments

- A tournament is a complete oriented graph.
- As a relational structure, it is a single trichotomous binary relation. ($x \rightarrow y$, $y \rightarrow x$ or $x = y$.)
- A competition between players: $x \rightarrow y$ means “$y$ wins.”
A **tournament** is a complete oriented graph.

As a relational structure, it is a single trichotomous binary relation. ($x \rightarrow y$, $y \rightarrow x$ or $x = y$.)

A **competition** between players: $x \rightarrow y$ means “$y$ wins.”

---

**Example**
Intervals and Simplicity

- An **interval**: set of points which “look” at every other point in the same way.
  - Synonyms: Autonomous sets, blocks, bound sets, closed sets, clumps, convex sets, modules...
  - A structure is simple if there are no proper intervals.
  - Synonyms: Indecomposable, prime...
An **interval**: set of points which “look” at every other point in the same way.

**Synonyms**: Autonomous sets, blocks, bound sets, closed sets, clumps, convex sets, modules...

- A structure is simple if there are no proper intervals.
- **Synonyms**: Indecomposable, prime...
Intervals and Simplicity

- An interval: set of points which “look” at every other point in the same way.
  - Synonyms: Autonomous sets, blocks, bound sets, closed sets, clumps, convex sets, modules...

- A structure is **simple** if there are no proper intervals.
  - Synonyms: Indecomposable, prime...
Intervals and Simplicity

- An interval: set of points which “look” at every other point in the same way.
- Synonyms: Autonomous sets, blocks, bound sets, closed sets, clumps, convex sets, modules...
- A structure is **simple** if there are no proper intervals.
- Synonyms: Indecomposable, prime...
Permutation $\pi$.

An interval of $\pi$ is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.
Intervals

- Permutation $\pi$.
- An interval of $\pi$ is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.

Example

[Graph showing a set of points in a Cartesian plane, possibly illustrating the concept of intervals and simple structures.]
Intervals

- Permutation $\pi$.
- An interval of $\pi$ is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.
Intervals

- Permutation $\pi$.
- An **interval** of $\pi$ is a set of contiguous indices $I = [a, b]$ such that $\pi(I) = \{\pi(i) : i \in I\}$ is also contiguous.
Simple Permutations

Only intervals are **singletons** and the **whole thing**.

Example
Only intervals are **singletons** and the **whole thing**.
Only intervals are singletons and the whole thing.
Only intervals are *singletons* and the *whole thing*.
Simple Permutations

- Only intervals are **singletons** and the **whole thing**.
Only intervals are \textit{singletons} and the \textit{whole thing}.

Example
Simple Permutations

- Only intervals are singletons and the whole thing.
Simplicity in Relational Structures

Concepts

Intervals and Simplicity

Simplicity in Graphs

- **Simple** graph?

Example

- Same neighbourhood $\Rightarrow$ interval.
Simple graph? Well, rather an *indecomposable graph*.

**Example**

- Same neighbourhood = interval.
Simple graph? Well, rather an indecomposable graph.

Example

Same neighbourhood = interval.
Simplicity in Graphs

- Simple graph? Well, rather an indecomposable graph.

Example

- Same neighbourhood $\Rightarrow$ interval.
Outline
Decomposing Permutations

- Every permutation has a block decomposition.
- Gives a unique simple permutation.

Example
Decomposing Permutations

- Every permutation has a block decomposition.
- Gives a unique simple permutation.

Example
Decomposing Permutations

- Every permutation has a block decomposition.
- Gives a unique simple permutation.

Example
If simple has > 2 points then the blocks are unique.

This is called the substitution decomposition.
If simple has \( > 2 \) points then the blocks are unique.
This is called the substitution decomposition.
Non-uniqueness

- Block decomposition is not unique.
Non-uniqueness

- **Block decomposition** is not unique.
Non-uniqueness

- Block decomposition is not unique.
Non-uniqueness

- Underlying structure is an increasing permutation.
Non-uniqueness

- Underlying structure is an increasing permutation.
Non-uniqueness

- Increasing permutation: both linear orders agree.
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

- Graphs: degenerate only (complete or independent graph).
- Posets: linear (linear order).
- Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

Graphs: degenerate only (complete or independent graph).

Posets: linear (linear order).

Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:

- **Degenerate** – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

- Graphs: degenerate only (complete or independent graph).
- Posets: linear (linear order).
- Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

- Graphs: degenerate only (complete or independent graph).
- Posets: linear (linear order).
- Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

**Graphs**: degenerate only (complete or independent graph).
- Posets: linear (linear order).
- Tournaments: linear only (transitive tournaments).
And in general...

- The substitution decomposition holds for every relational structure.
- Non-unique cases arise in two ways:
  - Degenerate – all relations complete or empty.
  - Linear – binary relations “agree”, others degenerate.
- Graphs: degenerate only (complete or independent graph).
- Posets: linear (linear order).
- Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

Graphs: degenerate only (complete or independent graph).

Posets: linear (linear order) or degenerate (antichain).

Tournaments: linear only (transitive tournaments).
The substitution decomposition holds for every relational structure.

Non-unique cases arise in two ways:
- Degenerate – all relations complete or empty.
- Linear – binary relations “agree”, others degenerate.

- Graphs: degenerate only (complete or independent graph).
- Posets: linear (linear order) or degenerate (antichain).
- Tournaments: linear only (transitive tournaments).
Eggs in a Basket

- **Structurally**, permutations, graphs, posets, tournaments, etc, belong to the same family.
Eggs in a Basket

- Structurally, permutations, graphs, posets, tournaments, etc, belong to the same family.
Counting Simples

- How many simple graphs are there?
- Asymptotically, almost all graphs are indecomposable.
- Also true for tournaments, posets, and single asymmetric relations.
Counting Simples

- How many simple graphs are there?
- Asymptotically, almost all graphs are indecomposable.
- Also true for tournaments, posets, and single asymmetric relations.
Counting Simples

- How many simple graphs are there?
- Asymptotically, almost all graphs are indecomposable.
- Also true for tournaments, posets, and single asymmetric relations.
How many simple *permutations* are there?

Asymptotically, only a few permutations are simple.

More precisely:

\[
\frac{n!}{e^2} \left( 1 - \frac{4}{n} + \frac{2}{n(n-1)} + O(n^{-3}) \right)
\]

(Albert, Atkinson and Klazar, 2003)
How many simple permutations are there?

Asymptotically, only a few permutations are simple.

More precisely:

\[
\frac{n!}{e^2} \left( 1 - \frac{4}{n} + \frac{2}{n(n-1)} + O(n^{-3}) \right)
\]

(Albert, Atkinson and Klazar, 2003)
Counting Simples

- How many simple permutations are there?
- Asymptotically, only a few permutations are simple.
- More precisely:

$$\frac{n!}{e^2} \left(1 - \frac{4}{n} + \frac{2}{n(n-1)} + O(n^{-3})\right)$$

(Albert, Atkinson and Klazar, 2003)
Useful or not?

- Approach questions about different relational structures in the same way.
- Should expect answers to be different.
Useful or not?

- Approach questions about different relational structures in the same way.
- Should expect answers to be different.