Modelling the Power Index Game

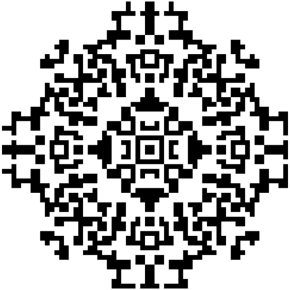
Charlie Gerrie

Dalhousie University

December 9th 2018

What will be discussed

- Introduction to the Power Index Game
- Subconfigurations Proportions
- Ratio Chains
- Global Structure



Framework

Definition

The set of *sides* $\mathbb{S}$ is the set containing two elements: strong ($\mathfrak{s}$) and weak ($\mathfrak{w}$).

Definition

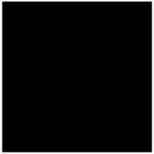
Every *game* $G(B, C)$ has two parts:

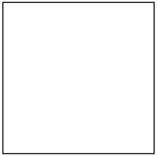
- A *board* B, which is a graph with vertex set V and an edge set.
- A *configuration* C, which is a function from V to $\mathbb{S}$.

A Note On Sides

Definition

The set of *sides* $\mathbb{S}$ is the set containing two elements: strong ($\mathfrak{s}$) and weak ($\mathfrak{w}$).

 = Strong

 = Weak

We will refer to these by their colour (i.e. black or white) from now on.

The Power Index

Definition

The *power* is determined by the following formula:

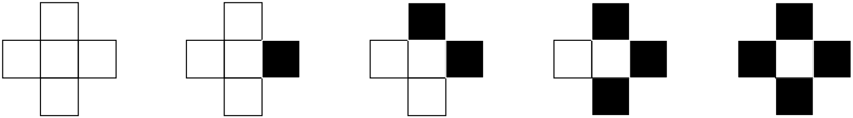
$$P(v) = \begin{cases} \frac{1}{|A|} & \text{if } |A| > |D|^* \\ 0 & \text{if } |D| > |A| \end{cases} \quad (1)$$

Where A is the set of vertices in the closed neighborhood agreeing, and D is the set of those disagreeing. * $\geq$ for strong

The Power Index

We limited ourselves to solely tori. Recall that a torus is essentially a grid which wraps on the edges. A torus is the cartesian graph product of two cycles.

Once we apply this simplificaton, there are only five cases for the neighborhood of a cell:



(a) $\frac{1}{5}$ *power*

(b) $\frac{1}{4}$ *power*

(c) $\frac{1}{3}$ *power*

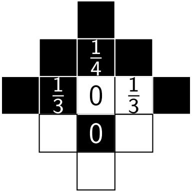
(d) 0 *power*

(e) 0 *power*

Figure: all possible neighborhoods for a center cell

Iterating

Once the powers have been calculated each cell looks in its closed neighborhood and chooses the most powerful cell. The cell will take on the colour of that strongest neighbor in the next iteration. Ties are broken in favor of not changing colour. A cell can be its own most powerful neighbor.



Cycles and Limiting Sets

Definition (Pre-cycle)

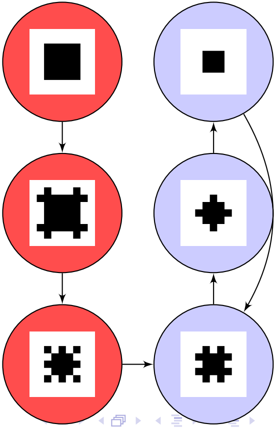
When iterating some starting configuration, the *pre-cycle* is the sequence of configurations that are never repeated. The red circles configurations on the right form a pre-cycle of length 3.

Cycles and Limiting Sets

Definition (Pre-cycle)

When iterating some starting configuration, the *pre-cycle* is the sequence of configurations that are never repeated. The red circles configurations on the right form a pre-cycle of length 3.

Figure: A 5-by-5 black square on a 9-by-9 board



Cycles and Limiting Sets

Definition (Limit-cycle)

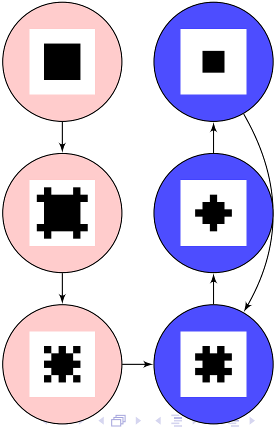
When iterating some starting configuration, the *limit-cycle* is the sequence of configurations that *are* repeated. The blue circles configurations on the right form a limit-cycle of length 3.

Cycles and Limiting Sets

Definition (Limit-cycle)

When iterating some starting configuration, the *limit-cycle* is the sequence of configurations that *are* repeated. The blue circles configurations on the right form a limit-cycle of length 3.

Figure: A 5-by-5 black square on a 9-by-9 board

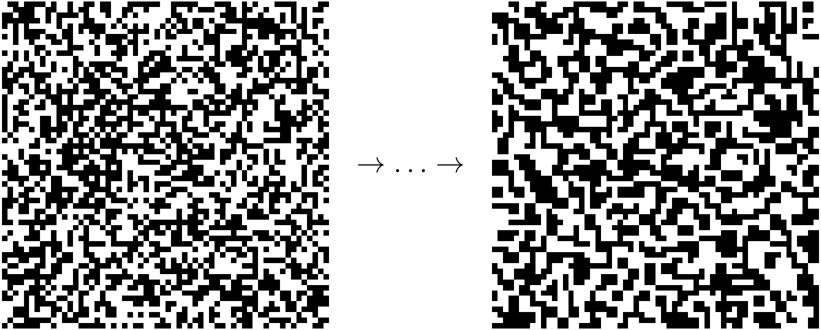


Some Things You Might've Expected

- *A maximum limit-cycle size.* Though the board size does put a bound on limit-cycle length, a "glider" exists, so arbitrarily large boards give arbitrarily large limit-cycles.
- *Power monotonically increases.* This is usually observed, but it is not a rule.

What We've Been Looking At

What we really care about is the large-scale long-term behavior. To do this, we focused on limit-cycle configurations. We used extensive computer simulations to do this.



Subconfigurations

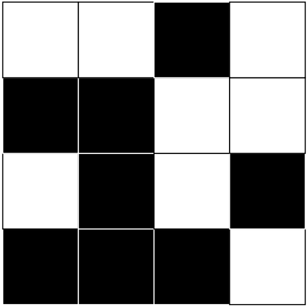
Definition

A *subconfiguration* is a configuration of a subgraph of the board of a game.

Subconfigurations

Definition

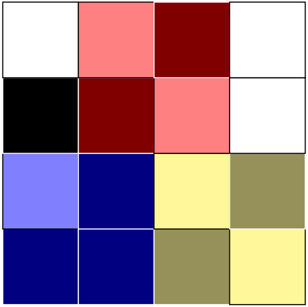
A *subconfiguration* is a configuration of a subgraph of the board of a game.



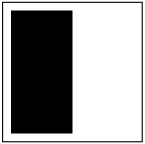
Subconfigurations

Definition

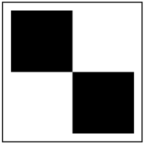
A *subconfiguration* is a configuration of a subgraph of the board of a game.



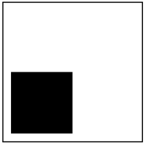
What We Are Counting



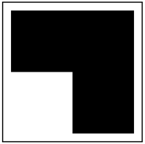
(a) Side



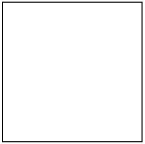
(b) Diagonal



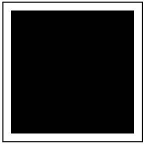
(c) White Corner



(d) Black Corner



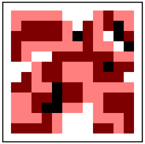
(e) White Square



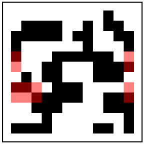
(f) Black Square

Figure: 2-square states

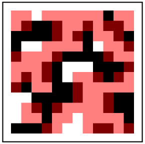
Counting Them



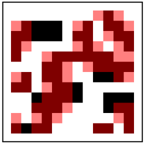
(a) Sides



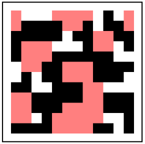
(b) Diagonals



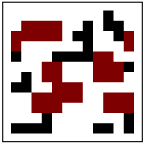
(c) White Corners



(d) Black Corners



(e) White Squares

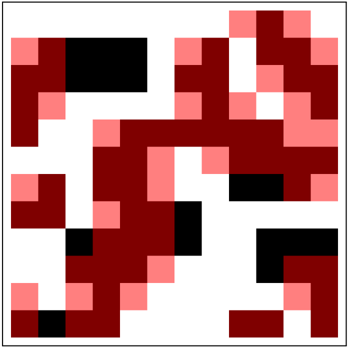


(f) Black Squares

Figure: 2-square samples from a 12-by-12 torus simulation

A Closer Look

- Note the overlapping
- How many do you think there are?
- Is there any rhyme or reason to where they are?



(a) Black Corners

Subconfiguration Proportions

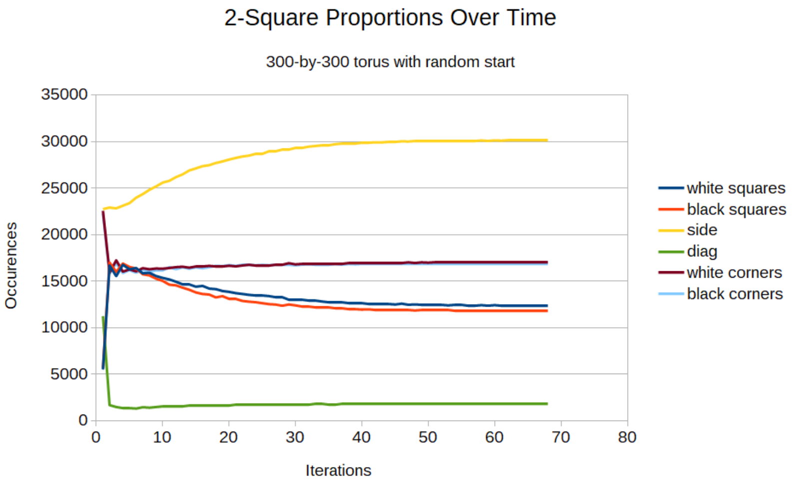
Definition

Representing the proportions of various subconfigurations as a tuple. Analogous to a probability vector for which subconfiguration a randomly chosen subconfiguration would be.

Example

A 10-by-10 torus has 37 sides, 12 diagonals, 2 white corners, 3 black corners, 23 white squares, and 23 black squares. We can represent this as (0.37,0.12,0.02,0.03,0.23,0.23). It is imperative that we remember the order of the tuple.

A Graph With A Lot To It



Ratio Chains

Definition

A *ratio chain* is a Markov Chain used to model subconfiguration proportions over time.

The goal is to be able to accurately predict how subconfiguration proportions change over time. A Markov Chain would be ideal since it condenses the game down to just its aggregate behavior.

Ratio Chains

A Markov Chain, and thus a Ratio chain, can be represented in two ways:

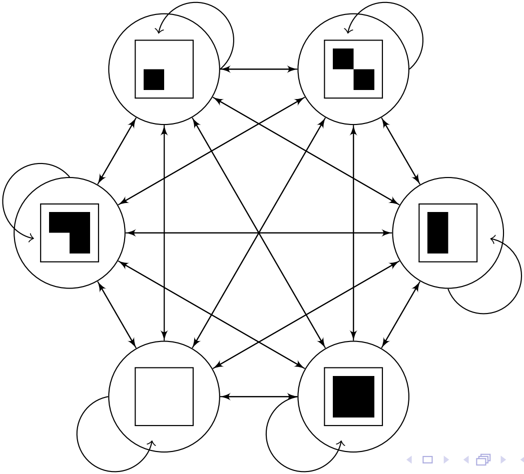
A matrix equation:

$$\vec{s}' = R\vec{s}$$

Such that $R_{i,j}$ is the probability of going from subconfiguration i to j.

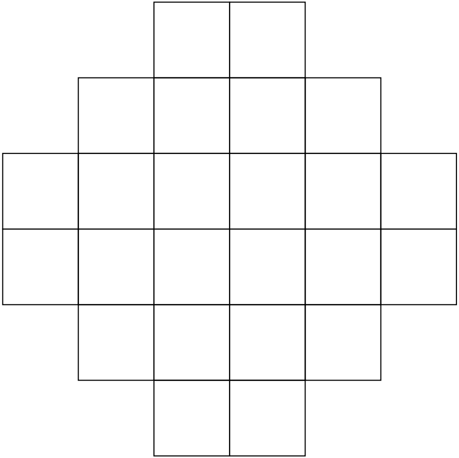
Ratio Chains

Or a graph, with the weights on the edges being the probabilities.



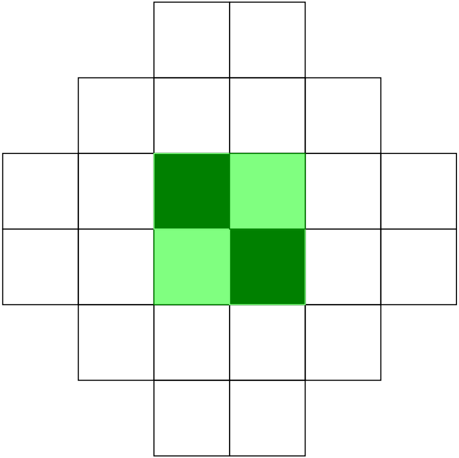
Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods



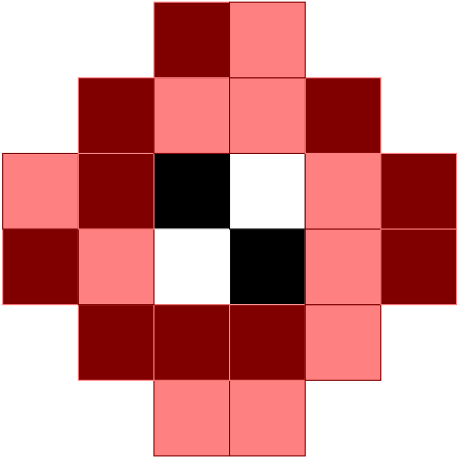
Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods



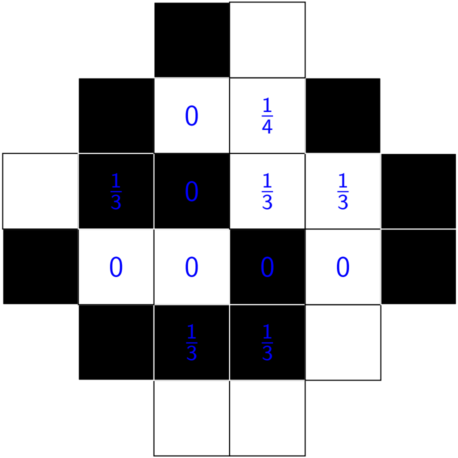
Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods



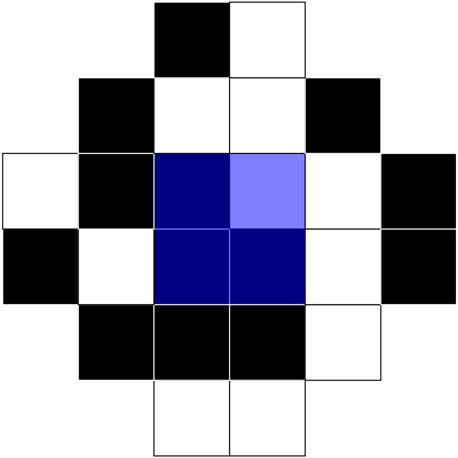
Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods



Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods



Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods

from, to	w. squares	b. squares	sides	diags	w. corners	b. corners
w. square	854756	228	21136	6856	3440	162160
b. square	228	854756	21136	6856	162160	3440
side	290368	290368	2696960	5760	455424	455424
diag	220786	220786	411592	241892	501048	501048
w. corner	20224	1769440	532928	24288	1709920	137504
b. corner	1769440	20224	532928	24288	137504	1709920

Constructing The First One

1. Pick and record a center
2. Pick and record a neighborhood
3. Iterate center once
4. Repeat for all centers and neighborhoods

from, to	w. squares	b. squares	sides	diags	w. corners	b. corners
w. square	0.815	0.000	0.020	0.006	0.003	0.155
b. square	0.000	0.815	0.020	0.006	0.155	0.003
side	0.069	0.069	0.643	0.001	0.108	0.108
diag	0.105	0.105	0.196	0.115	0.239	0.239
w. corner	0.005	0.422	0.127	0.006	0.408	0.033
b. corner	0.422	0.005	0.127	0.006	0.033	0.408

Using It

$$\begin{bmatrix} (\texttt{white squares})' \\ (\texttt{black squares})' \\ (\texttt{side})' \\ (\texttt{diag})' \\ (\texttt{white corner})' \\ (\texttt{black corner})' \end{bmatrix} = \begin{bmatrix} 0.815 & 0.000 & 0.069 & 0.105 & 0.005 & 0.422 \\ 0.000 & 0.820 & 0.069 & 0.105 & 0.422 & 0.005 \\ 0.020 & 0.020 & 0.643 & 0.196 & 0.127 & 0.127 \\ 0.006 & 0.006 & 0.001 & 0.115 & 0.006 & 0.006 \\ 0.003 & 0.155 & 0.108 & 0.239 & 0.408 & 0.033 \\ 0.155 & 0.003 & 0.108 & 0.239 & 0.033 & 0.408 \end{bmatrix} \begin{bmatrix} (\texttt{white squares}) \\ (\texttt{black squares}) \\ (\texttt{side}) \\ (\texttt{diag}) \\ (\texttt{white corner}) \\ (\texttt{black corner}) \end{bmatrix}$$

Using It

$$\begin{bmatrix} 0.251 \\ 0.018 \\ 0.177 \\ 0.177 \\ 0.188 \\ 0.188 \end{bmatrix} = \begin{bmatrix} 0.815 & 0.000 & 0.069 & 0.105 & 0.005 & 0.422 \\ 0.000 & 0.820 & 0.069 & 0.105 & 0.422 & 0.005 \\ 0.020 & 0.020 & 0.643 & 0.196 & 0.127 & 0.127 \\ 0.006 & 0.006 & 0.001 & 0.115 & 0.006 & 0.006 \\ 0.003 & 0.155 & 0.108 & 0.239 & 0.408 & 0.033 \\ 0.155 & 0.003 & 0.108 & 0.239 & 0.033 & 0.408 \end{bmatrix} \begin{bmatrix} 0.0625 \\ 0.0625 \\ 0.250 \\ 0.125 \\ 0.250 \\ 0.250 \end{bmatrix}$$

This correctly predicts the proportions for the first iteration of a torus with a random starting proportion.

Limitations

Shape	Actual Starting	Actual Final	Ratio Chain Final	Final Error
Side	0.250	0.334	0.122	-0.212
Diagonal	0.125	0.020	0.006	-0.014
White Corner	0.250	0.188	0.116	-0.072
Black Corner	0.250	0.188	0.116	-0.072
White Square	0.062	0.135	0.320	0.185
Black Square	0.062	0.135	0.320	0.185

Unfortunately, this chain does not correctly predict long-term proportions. It appears to *overshoot* its target.

Trying to Construct a Better One

We have tried several approaches to improve the chain.

- Limiting the neighborhoods included to only those within a confidence interval of the actual long-term proportions.
- Weighing the contribution of each neighborhood by $e^{-||p-\hat{p}||}$
- Simulating long-term grids and randomly sampling actual transitions.

Sadly, all of these had underwhelming results.

Trying to Construct a Better One

Shape	Actual Starting	Actual Final	Original Chain	Original Error	New Chain	New Error
Side	0.250	0.334	0.122	-0.212	0.018	-0.316
Diagonal	0.125	0.020	0.006	-0.014	0.115	0.095
White Corner	0.250	0.188	0.116	-0.072	0.033	-0.155
Black Corner	0.250	0.188	0.116	-0.072	0.034	-0.154
White Square	0.062	0.135	0.320	0.185	0.394	0.259
Black Square	0.062	0.135	0.320	0.185	0.406	0.271

Global Structure

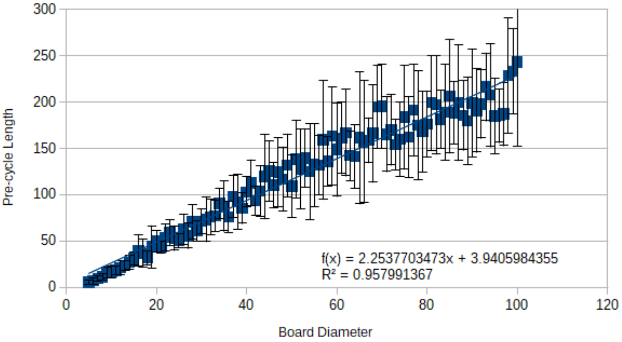
Definition (Local Behavior)

Behavior that only depends on a bounded neighborhood.

It appears that there is non-local behavior, or global behavior, to the power index game.

Global Structure

This is a graph of the average pre-cycle length for random starting configurations on boards of sizes varying from 5-by-5 to 100-by-100.



Conclusion

We started using ratio chains with the intention of being able to use them to model the Power Index Game in a simplified way. They have had some success at that, specifically within the first iteration. Howevever, long-term modelling still seems to be out of reach.

There appears to be an argument for global structure. Thus, I support further study of the aggregate behavior of the game.

Thank You and Acknowledgments

- **Jeannette Janssen** and **Jordan Barrett** for their supervision and guidance.
- **NSERC** for making this all possible through their funding.
- **Dalhousie**