


Fall 2018

Supersymmetry and the Math of Adinkras

Ari Putt Spiesberger
Bard College

Follow this and additional works at: https://digitalcommons.bard.edu/senproj_f2018

 Part of the [Elementary Particles and Fields and String Theory Commons](#), and the [Other Applied Mathematics Commons](#)



This work is licensed under a [Creative Commons Attribution-Noncommercial-No Derivative Works 4.0 License](#).

Recommended Citation

Spiesberger, Ari Putt, "Supersymmetry and the Math of Adinkras" (2018). *Senior Projects Fall 2018*. 41. https://digitalcommons.bard.edu/senproj_f2018/41

This Open Access work is protected by copyright and/or related rights. It has been provided to you by Bard College's Stevenson Library with permission from the rights-holder(s). You are free to use this work in any way that is permitted by the copyright and related rights. For other uses you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/or on the work itself. For more information, please contact digitalcommons@bard.edu.

Supersymmetry and the Math of Adinkras

A Senior Project submitted to
The Division of Science, Mathematics, and Computing
of
Bard College

by
Ari Spiesberger

Annandale-on-Hudson, New York
December, 2018

Abstract

The purpose of this paper is to expand the dictionary of values related to parameterized supersymmetry values. These values, represented by Adinkras, are some of the most fascinating explanations of theoretical supersymmetry that exist. My goal was to approach and define an equivalence class on a specific value that had yet to be defined. I was able to do this, and in doing so, present information on a larger equivalence class in the field surrounding Adinkras.

Contents

Abstract	iii
Dedication	vii
Acknowledgments	ix
1 Introduction	1
2 The Physics Behind it All	5
2.1 One-Dimensional Parameterization	6
2.2 Clifford algebra and gamma matrices	7
2.3 Chiral Multiplet	8
2.4 Vector Supermultiplets	13
2.4.1 Base Change	17
3 A small chapter on graph theory	19
3.1 The Adinkra	20
3.1.1 Nodes	20
3.1.2 Edges	20
3.2 Adinkra Graph Theory Properties	21
3.3 Adinkra Identities	23
4 Quotenting and the Valise	27
4.1 The Adjusted Adjacency Matrix	29
4.2 The L matrix	33
4.3 Valise and Matrix Representation	34
4.4 The Node Flip	35
4.5 The Node Swap	37
4.6 Valise Matching	39

5	My Results, Specific Values of Adinkras	43
5.1	$\frac{1}{2}(L_i R_j - L_j R_i)$	43
5.2	Possible Permutations	44
5.3	The Permutation Matrix Group	45
5.4	The Permutation Form $L_i R_j$	46
5.5	The permutation of $\frac{1}{2}(L_i R_j - L_j R_i)$	47
5.6	First Big Proof	48
5.7	The Bigger Proof	50
5.8	Fermion Invariance	51
5.9	Future Work	52

Dedication

Dedicated to Bard College. An absolutely crazy and wonderful place.

Acknowledgments

This project is dedicated to my Parents, John and Mary, whose love and value for knowledge is the greatest gift I have received. To the Bard Math department, and the professors whose passion for mathematics is something they bring to each and every one of there classes. To Stefan Mendez-Diez, who took a tremendous amount of time to teach me every little detail that I needed in order to even begin working on this project. Finally, to my friend Bill.

1

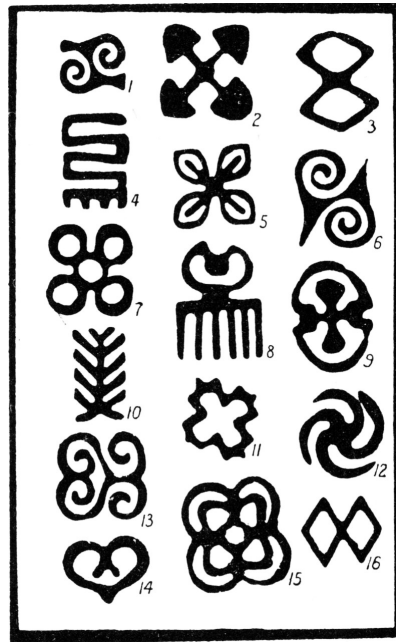
Introduction

Humans have sought symmetry in the world since the existence of the species. Most of how humans interact with the world is based on observation of symmetry or lack thereof. As hunters we sought disturbances in the symmetry of nature that would signify prey or predator. As farmers we strive to create symmetry in crops to maximize efficiency of space. In mathematics we build symmetric models, and seek lack of symmetry to give us clues to what we do not yet know.

As physicists look at the universe, symmetry offers beautiful and elegant relationships that make very difficult problems simpler. This symmetry in the universe is appealing as humans themselves find symmetry fascinating and beautiful. It is possible that this is why physics is so concerned, especially in theory, in symmetric models of complex structures. Supersymmetry is just one of these models.

Preceding supersymmetry, there were many theories in physics proposing symmetries between particles. Often these were incomplete models as it does not give a complete view of the particles in systems. Generally limitations were fermions were not well represented in these models. Supersymmetry in a modern form was first proposed by Hironari Miyazawa in 1966 [12], and has been developed deeply through the course of the 50 years leading us to today. Supersymmetry

Figure 1.0.1. Example of west african Adinkra symbols



still very much exists in a theoretical space. The transformations proposed by supersymmetry require very high energy levels, possibly higher than even seen in the Large Hadron Collider, and so we have not seen much direct evidence for the validity of Supersymmetry. Despite this, many physicists and mathematicians are very interested in these theories, and continue to develop it.

In an article proposed by Professors Jim Gates Jr, (University of Maryland) and Micheal Faux, (Suny Albany), a new way of dealing with supersymmetry was developed. This idea was representing supersymmetry with an Adinkra [?, ?]. An Adinkra is “a symbol that represents concepts or aphoris” (Wikipedia, Adinkra). Adinkras originated in crafts and textiles from west africa. These Adinkras were used by Gates and Faux to build a graphical representation of these very complicated supersymmetry equations in order to simplify the problems in supersymmetry. The long term goal is that the graphical representation given by Adinkras will help us understand how supersymmetry would work in a very general way.

The goal of my work and research was to determine the relationship between a specific value of Adinkras and its cousin permutations and negations. I was able to determine a previously unknown relationship (hopefully!), and will continue to try to determine more information about these relationships. The next step of this project is to attempt to continue to develop it and publish it as a stand alone piece of work, or attach it to one of my advisors papers. It has been an absolute pleasure to be involved in this work.

2

The Physics Behind it All

We are using Adinkras to try to help our understanding of the “Off-Shell Supersymmetry problem”. We seek to explore the algebra

$$\{Q_I, Q_J\} = 2i\delta_{IJ}\partial_\mu$$

Where $\partial_\mu = (\partial_\tau, \partial_x, \partial_y, \partial_Z)$

This is the supersymmetry algebra in 4 dimensions. When we work with Adinkras we are doing a reduction from 4 dimensions to 1 dimension. This means that the gradient ∂_μ , must be reduced to a one-dimensional form. The equation for ∂_μ has four parts, partials with respect to time, and three “spatial” dimensions. In this case we choose to work with a derivative with respect to time. This is the framework of the 0-brane reduction.

With Q 's being supercharges, “that act non-trivially on both propagating and auxiliary fields” [?, ?]. From this algebra we can generate many of the Adinkras with a number of reductions. In this project, I have done a number of these reductions, on a (0-brane) and a vector multiplet. Included will be those reductions as well as how those reductions are applied to the adinkras we work on.

I will later explain the math, but it turns out that when we reduce a susy algebra from four dimensions to one, we get this result. $\partial_\mu = \lambda_\mu \partial_\tau$ (See section 2.2, one dimensional parameterization)

The general equations for the Adinkras will then be...

$$\{Q_I, Q_J\} = 2i\delta_{IJ}\partial_\tau$$

Where the ∂_μ from the first equation has been parameterized to ∂_τ .

2.1 One-Dimensional Parameterization

In order to move from 4 dimensions to 1 dimension, we proceed as follows: four dimensional space is projected onto the one-dimensional line parameterized by:

$$\lambda_\mu = \cos \alpha T_\mu + \sin \alpha \sin B \cos \gamma X_\mu + \sin \alpha \sin B \sin \gamma Y_\mu + \sin \alpha \cos B Z_\mu$$

Here X_u, Y_u and Z_u are spacial dimension gradiants. T_u is the time dimension component.

For our particular purposes we choose a parameterization to be solely in the time direction.

To do this, we choose to set $\alpha = 0$ and this results in...

$$\lambda_\mu = T_\mu = (1, 0, 0, 0)$$

and when we do the dot product of the gamma matrices with the lambda parameterization we get

$$\gamma^\mu \cdot \lambda_\mu = \gamma^\mu \cdot (1, 0, 0, 0) = \gamma^0$$

2.2 Clifford algebra and gamma matrices

In supersymmetry fermions are represented with spinor fields. To represent this we need use the Clifford Algebra as it has the anti-commuting property we need for spinors. There are 4 original gamma matrices that I am addressing in this paper. We have $\gamma^0, \gamma^1, \gamma^2, \gamma^3$. We also have $\gamma^5 = i\gamma^0\gamma^1\gamma^2\gamma^3$ These matrices anti-commute, and this ensure that they generate a matrix representation of the Clifford Algebra.

It is very important to note that these gamma matrices can have many different representations, depending on which basis is selected. I will stick with the basis I give chosen to represent susy algebra in this paper. The properties remain the same, and this is what is important. We need to follow the Clifford Algebra, and the rules that come with it. The gamma matrices ensure this.

The transformations used to create Adinkras require specific use of gamma matrices. By plugging into specific equations we can do transformations into just one dimension, the time dimension. One type of this reduction is called the 0-brane reduction. There are a number of different transformations we can do on the susy equations.

To do this will have to define gamma matrices which will remain the same through all equations.

Note: σ^u corresponds to the pauli matrices.

These gamma matrices are as follows, as defined for our susy models by Professor Gates. [9]

$$(\gamma^0)_b^a = i(\sigma^3 \otimes \sigma^2)_a^b = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$\begin{aligned}
(\gamma^1)_b^a &= (I_2 \otimes \sigma^1)_a^b = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\
(\gamma^2)_b^a &= (\sigma^2 \otimes \sigma^2)_a^b = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} \\
(\gamma^3)_b^a &= (I_2 \otimes \sigma^3)_a^b = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
\gamma^5 &= -(\sigma^1 \otimes \sigma^2)_a^b = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}
\end{aligned}$$

2.3 Chiral Multiplet

“The $4D$, $\mathcal{N} = 1$ Chiral Multiplet is very well known to consist of a scalar A , a pseudoscalar B , a Majorana fermion a , a scalar auxiliary field F , and a pseudoscalar auxiliary field G . A convenient way to express the supersymmetry variation of these component fields is by first regarding them as the lowest component of a superfield (denoted by the same symbol) and then expressing the action of the superspace covariant derivative D_a acting on each. As we have included the auxiliary fields F and G , necessarily it is the off-shell theory under consideration.”(6, Gates, Geonomics).

We have 4 equations we build these matrices from.

1. $D_a A = \psi_a$
2. $D_a B = i(\gamma^5)_a^b \psi_b$
3. $D_a F = (\gamma \cdot \lambda)_a^b d_t \psi_b$
4. $D_a G = i(\gamma^5 \gamma \cdot l)_a^b d_t \psi_b$

For equation 1) We need to simply define the indices. This means that our a indices are assigned values on both sides of the equation. This makes these sets of equations very simple and easy to work on. For equation 1:

$$D_1 A = \psi_1$$

$$D_2 A = \psi_2$$

$$D_3 A = \psi_3$$

$$D_0 A = \psi_0$$

For the next set of equations we match our a indices, and our b indice corresponds to the row in γ^5 that indice a selects.

$$D_0 B = i\psi_4$$

$$D_1 B = -i\psi_3$$

$$D_2 B = i\psi_2$$

$$D_3 B = -i\psi_1$$

In the next equation we have the ∂_τ , which means we need to do a reduction from 4 dimensional space to 1 dimensional space we discussed at the beginning of this chapter. Again our λ is:

$$\lambda = \cos(\alpha)\tau_\mu + \sin \alpha \sin \beta \cos \gamma X_\mu + \sin \alpha \sin \beta \sin \gamma \Gamma_\mu + \sin \alpha \cos \beta Z_\mu$$

Each gamma and lambda function has a part in equation 3, γ^μ, λ_μ , is associated together in the dot product. So for each matrix, we get the lambda piece with it.

These are each the reperamaterizations of the line in 1 dimension along each of particular one

dimensional directions, which were originally framed in the ∂_μ gradient.

$$\lambda_0 = p = \cos(\alpha)\tau_\mu$$

$$\lambda_1 = z = \sin(\alpha)\sin(\beta)\cos(\gamma)X_\mu$$

$$\lambda_2 = w = \sin(\alpha)\sin(\beta)\sin(\gamma)\Gamma_\mu$$

$$\lambda_3 = f = \sin(\alpha)\cos(\beta)Z_\mu$$

$$\gamma^0\lambda_0 = \begin{bmatrix} 0 & p & 0 & 0 \\ -p & 0 & 0 & 0 \\ 0 & 0 & 0 & -p \\ 0 & 0 & p & 0 \end{bmatrix}$$

$$\gamma^1\lambda_1 = \begin{bmatrix} 0 & z & 0 & 0 \\ z & 0 & 0 & 0 \\ 0 & 0 & 0 & z \\ 0 & 0 & z & 0 \end{bmatrix}$$

$$\gamma^2\lambda_2 = \begin{bmatrix} 0 & 0 & 0 & -w \\ 0 & 0 & w & 0 \\ 0 & w & 0 & 0 \\ -w & 0 & 0 & 0 \end{bmatrix}$$

$$\gamma^3\lambda_3 = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & -f & 0 & 0 \\ 0 & 0 & -f & 0 \\ 0 & 0 & 0 & f \end{bmatrix}$$

We have to add all of these together to continue to develop the computations, and we get a final matrix. Reading across, we can get values for each equation,

$$D_0F = f\psi_0 + (p+z)\psi_1 - w(\psi_3)$$

$$D_1F = (-p+z)\psi_0 + f\psi_1 + w\psi_2$$

$$D_2F = w\psi_1 - f\psi_2 + (-p+z)\psi_3$$

$$D_3F = -w\psi_0 + (p+z)\psi_2 + f\psi_2$$

In order for us to reduce to the 0 brane, we reduce everything down to $\alpha = 0$ from our gauge.

In the the lambda equations, this sets everything but $\lambda_0 = 1$, and all others to 0. This reduction

looks like.

$$D_0 F = \psi_1$$

$$D_1 F = -\psi_0$$

$$D_2 F = -\psi_3$$

$$D_3 F = \psi_2$$

Each gamma and lambda function has a part, γ^u, λ^u , is associated together in the dot product. So for each matrix, we get a lambda piece with it. $\lambda_0 = \cos\alpha$ as each other piece of the lambda function goes to 0. So for each of these γ^1, λ_1 are the set dot product.... and so on and so forth.

The final product is $D_a G = i(\gamma^5 \gamma \cdot \lambda)_a^b \delta_\tau \psi_b$

Again here, we have a reduction from 4 dimensions onto 1 dimension, specifically onto ∂_τ , which means that λ requires that ∂_τ is the only element which is satisfied by $\alpha = 0$. This sets out $(\gamma^5 \gamma)$ to be only $(\gamma^5 \gamma^0)$

We have to compute each product matrix, which gives us

$$\begin{aligned}
 (\gamma^5 \gamma^0) &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \\
 (\gamma^5 \gamma^1) &= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} \\
 (\gamma^5 \gamma^2) &= \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 (\gamma^5 \gamma^3) &= \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}
 \end{aligned}$$

Again we just pair off a lambda to each of the gamma product matrices. The summation of these matrices adds up to,

$$\begin{bmatrix} -w & 0 & (p+z) & -f \\ 0 & w & -f & (p-z) \\ (-p+z) & -f & w & 0 \\ -f & (-p-z) & 0 & w \end{bmatrix}$$

and the individual computations just take the summation of each row. I will display the row and then the 0-brane reduction. This reduction occurs along the same parameters through the paper. In order to reduce ∂_μ to ∂_τ we can only have λ_0 which sets γ to γ^0 . Therefore, like the above equations, we can remove everything but p .

$$D_0G = -wi\psi_0 + i(p+z)\psi_2 - if\psi_3$$

And by substituting back from the previous equations.

$$D_0G = -i \sin(\alpha) \sin(\beta) \sin y \psi_0 + 0 + i(\cos(\alpha) + \sin(\alpha) \sin(B) \cos(y))$$

reduces to $i\psi_2$

$$D_1G = -iw\psi_1 + if\psi_2 + i(p-z)\psi_3$$

reduces to $i\psi_3$

$$D_2G = i(-p+z)\psi_0 - if\psi_1 + w\psi_2 + 0$$

reduces to $-i\psi_0$

$$D_3G = -if + i(-p-z) + iw$$

reduces to $-i\psi_1$

4 sets of results can reduce down to a singular matrix, as an expression of rows from ψ_a and columns from D_aA, B, F, G

These matrices are,

$$D_aA = I_4$$

$$D_aB = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$D_a F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D_a G = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

Each of these matrices gives us the information on the action of 4 transformations on a particle represented by A, B, F, G . This paper is more interested in the matrix transformations themselves. By taking the list of entries from the first through fourth row we can build these new matrices. They are represented as such.

$$L_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$L_3 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$L_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

These are the matrix representations from particle to particle according to our susy physics for the chiral multiplet.

2.4 Vector Supermultiplets

We want to do a similar analysis on the vector supermultiplets

Starting with $\partial_\mu = \lambda_\mu \partial_\tau$

The starting equations for the vector supermultiplet are, (λ_b is a fermion)

1) $i(\gamma^5 \gamma^\mu)_a^b \partial_\mu \lambda_\beta = i(\gamma^5 \gamma^\mu)_a^b \lambda_\mu \partial_\tau \lambda_\beta$ Because the reduction sets $\alpha = 0$ in the lambda equation and the $i(\gamma^5 \gamma^\mu \lambda_\mu)$

2) $(\gamma_i)_a^b \psi_b$ is selected by the specific gage we choose for this reduction which is the calumn gage. This sets $i = 1, 2, 3$ to the appropriate gamma matrice matched to the equation, and for $i = 0$ we get all an equation that reduces to 0. Because in the lambda, we choose to set These vector supermultiplets are defined by the equations as given, I wish to sort out the defined relationships, similar to the one from above.

After the 0 brane reduction we get these equations for the bosons.

$$\begin{aligned} 1. D_a A_i &= (\gamma_i)_a^b \psi_b \\ 2. D_a d &= i(\gamma^5 \gamma^\mu T_\mu)_a^b \delta_b \end{aligned}$$

And for the fermions

$$3. D_a \delta_b = -\frac{i}{2}([\gamma \cdot \tau, \gamma^i]_{ab} (\partial_\tau A_i) + (\gamma^5)_{ab} (\partial_\tau d))$$

Tau in this comes from the already broken down equations from above. We set α to be 0, and $\tau = (1, 0, 0, 0)$.

Again the calumn gage selects only $i = 1, 2, 3$

$$1a. D_a A_1 = (\gamma^1)_a{}^b \psi_b$$

$$1b. D_a A_2 = (\gamma^2)_a{}^b \psi_b$$

$$1c. D_a A_3 = (\gamma^3)_a{}^b \psi_b$$

We can quickly compute $D_0 A_i$ terms from our gamma matrices we listed above.

$$\begin{aligned} D_0 A_1 &= \psi_2 & D_1 A_1 &= \psi_1 & D_2 A_1 &= \psi_4 & D_3 A_1 &= \psi_3 \\ D_0 A_2 &= -\psi_4 & D_1 A_2 &= \psi_3 & D_2 A_2 &= \psi_2 & D_3 A_2 &= -\psi_1 \\ D_0 A_3 &= \psi_1 & D_1 A_3 &= -\psi_2 & D_2 A_3 &= \psi_3 & D_3 A_3 &= -\psi_4 \end{aligned}$$

So now we have the $D_a d$ equation to operate on.

We need to solve this by computing $(\gamma^5 \gamma^\mu \tau_\mu)$

The only non-zero matrix here is if the selected gamma matrix is γ^0

This allows us to compute $i(\gamma^5 \gamma^0)$

We now need to do matrix multiplication across the gamma matrices,

$$i(\gamma^5 \gamma^0) = i \begin{bmatrix} 0 & 0 & i & 0 \\ 0 & 0 & 0 & i \\ -i & 0 & 0 & 0 \\ 0 & -i & 0 & 0 \end{bmatrix}$$

This gives us these equations...

$$D_0 d = -\psi_3$$

$$D_1 d = -\psi_4$$

$$D_2 d = \psi_1$$

$$D_3 d = \psi_2$$

We have now a new set of matrices that represent the operations of these actions. These matrices are again L_1, L_2, L_3, L_4

$$L_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

$$L_3 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$L_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

These four matrices represent the actions from bosons to fermions from the vector multiplet.

The third equation at the beginning of the Chapter,

$$D_a \delta_b = -\frac{i}{2}([\gamma \cdot \tau, \gamma^i]_{ab}(\partial_\tau A_i) + (\gamma^5)_{ab}(\partial_\tau d))$$

This equation maps the fermions to bosons, and we could solve, which is in the appendix. However It is simpler to note that, for now, these matrices will simply be the transpose of the first we created.

2.4.1 Base Change

In my work, I decided that the results coming from the Chiral and Vector multiplets were over-complicated. By changing the basis for the matrices, I can rewrite both the Chiral and Vector multiplets in a more convenient form. This is the form that I chose to stick with throughout the course of my research. This is also the standard form used in papers today to represent these specific multiplets.

$$\text{The Vector Multiplet is... } D_0 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, D_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D_2 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}, D_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

The Chiral Multiplet is

$$D_a A = I_4$$

$$D_a B = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$D_a F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$D_a G = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

Here are the changes of the matrices that still satisfy the physics and naturally, the properties required for susy parameters.

3

A small chapter on graph theory

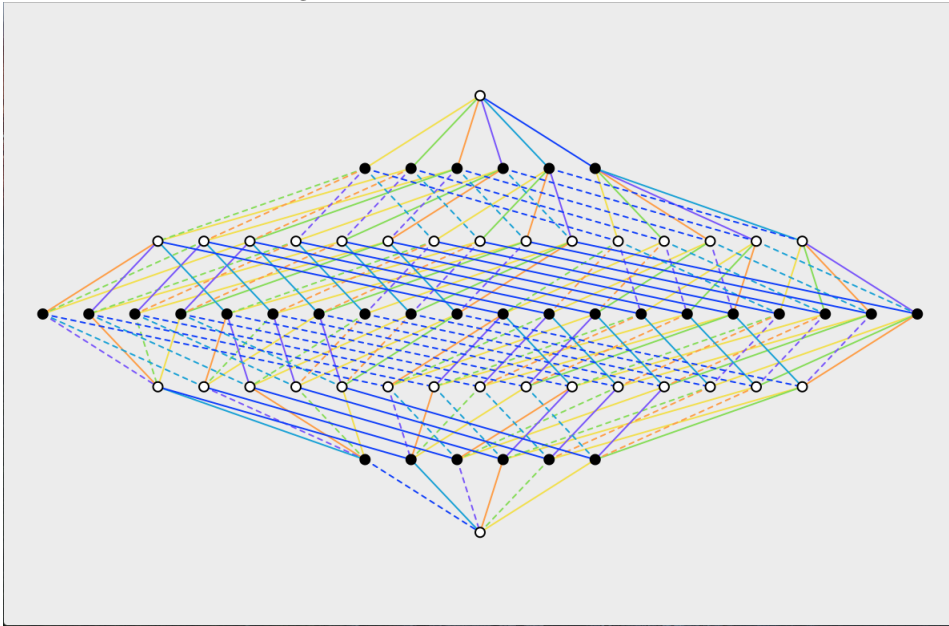
To begin this project, it is important to discuss, what is graph theory and how do we use it?

Graph theory is a branch of mathematics informally developed in 1736 by Leonard Euler. Since then it has proved itself a valuable tool that is used in many different applications. These graphs are mathematical structures used to define the relationships between objects. When we talk about Adinkras in this paper, we need to define a few terms.

The whole idea behind supersymmetry is that we have some energy state transformation between Bosons and Fermions. It is then a sensible approach to represent these transformations graphically. In a picture we can represent these transformations in a much simpler way than we would be able to with a list of many equations. The picture also allows us to get a much different understanding of the supersymmetry than we would get from just calculations.

In this chapter I will give terms to the graphs are Adinkras. Adinkras are the special graphs that we use to represent Supersymmetry. These Adinkras were created by Professor Jim Gates and Micheal Faux in 2004. [7]

Figure 3.0.1. 6 dimensional Adinkra



3.1 The Adinkra

What is a graph?

In mathematics a graph is a structure amounting to a set of objects in which some pairs of the objects are in some sense "related" [15]

In the following sections I will go about explaining and naming all of the components of the Adinkra graph in order to use common terminology throughout the paper.

3.1.1 Nodes

In my research, these nodes will represent the points on the graph. There are two separate kinds of nodes, bosons, and fermions. The bosons and fermions are colored differently in the Adinkra Graph. bosons are solid filled in points, and Fermions are unfilled points on this graph. Having two different sets of nodes is what we call Bipartite, in graph theory.

3.1.2 Edges

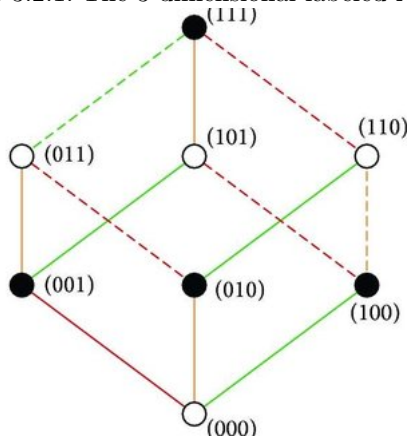
Between bosons and fermions we may have a colored line, These represent the susy (short for supersymmetry) transformations between a boson and a fermion. There are many different col-

ored edges, but each edge of the same color represents the same transformation mathematically.

Together the nodes and edges will create an Adinkra-which we use to represent the super-symmetry. The Adinkra is a graph that has a number of properties.I will give the reader a basic understanding of adinkras before diving deeper into the physics and math behind this project.

3.2 Adinkra Graph Theory Properties

Figure 3.2.1. The 3 dimensional labeled Adinkra



I will now go through some of the mathematical properties of the Adinkra. These terms are mostly terminology but have mathematical properties that we will use down the road.

Definition 3.2.1. Simple-Adinkras are simple: This means that no edges go from a node to itself. Another way of saying this is that we have no loops. Secondly for any two nodes, at most one edge connects them. No two edges connect the same node pair. Attached are some examples of simple graphs.

Wolfram Alpha has excellent visual representation of this concept.

Definition 3.2.2. Connected-The types of Adinkras we will be looking at are called connected graphs. A connected graph means that there is a path between any two nodes. This also means

that no node is unreachable from another node. It turns out that some Adinkras are not connected, but this occurs when we create a new Adinkra by "adding" two Adinkras together.

Definition 3.2.3. Bipartite-Bipartite graphs means that there are two distinct sets of nodes. These sets we defined earlier in this chapter as the set of bosons, written in as a solid point, and Fermions, the unfilled point. No two nodes in the same set (bosons, fermions) are connected by an edge. This means no edge goes from a boson to a boson, and no edge goes from a Fermion to a Fermion. There are certainly types of symmetries defined in physics that do this, but this is one of the main parts of Supersymmetry.

Definition 3.2.4. Finite-This name is self explanatory. These Adinkras do not contain infinite edges or nodes. We have a maximum of 32 dimensions in our graphs, which means that we have a limited number of possible nodes. 32 dimensions in these Adinkras means that we have 32 different types of edges., and since these graphs are connected, this means we have a finite number of nodes.

It turns out that the number of possible Adinkras is massive, at this point still uncounted. We do however know that these too are not possibly infinite.

Definition 3.2.5. Regular-Adinkras are regular, which means that each node (n) has exactly (i) edges incident to the node. (incident means attached).This is the case in all Adinkras. It also works out that each node (n) has only 1 of each color of edge incident to it.

Definition 3.2.6. Colored- Adinkras have colored edges, these edges will indicate which transformation action is occurring between boson and fermion. These edges are colored in different shades. The colours of the edges are represented by the transformations in the previous chapter on the physics. The physical equations give us a matrices that represent the action of certian transformations. The nodes are colored black and white to represent the bipartite structure.

Definition 3.2.7. Nodes-We label nodes with binary code of length N. When we label nodes, the length of each word is consistent in any Adinkra. We should consider these binary codes $\mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2 \dots \oplus \mathbb{Z}_2$ A three dimensional Adinkra will have nodes,

(000), (001), (010), (100), (110), (011), (101), (111). In an unquotiented Adinkra, the number of 1's in the binary determines the weight of the node, and we call this weight "odd" or "even". This all changes when we quotient the group of our nodes.

3.3 Adinkra Identities

Dimension-The dimension of an Adinkra is defined by the number of colored edges in the Adinkra. It is also equivalent to the amount of \mathbb{Z}_2 groups in the total group product. Mathematically, the dimension of the adinkra is the number of susy generators in the Adinkra. **Even and Odd Nodes**-Even Nodes in the Adinkra are the bosons. The Odd Nodes in the Adinkra are the Fermion Nodes. We always differentiate the Even and Odd sets as the bosons and fermions. This is how we mathematically differentiate between the different sets.

Here is a quick theorem to prove that there are an equal number of odd and even weighted nodes in any dimensional set we choose. This is a quick small direct proof I developed for the math majors out there.

Theorem 3.3.1. *The complete set of all values $\mathbb{Z}_2 \oplus \mathbb{Z}_2 \oplus \mathbb{Z}_2 \dots \oplus \mathbb{Z}_2$, has binary values with weight h . h is an integer, and therefore is either odd or even. If we call define a function $f(x) = x + (0000\dots001)$ where x is an even weighted binary code, we know that x has an even valued h . Since $(0000\dots0001)$, has weight 1, we $f(x)$ has weight $h+1$ if x has final value \mathbb{Z}_2 value 0, and $f(x)$ has weight $h-1$ if x has final value \mathbb{Z}_2 has value 1. Therefore $f(x)$ has an odd weight. $f(x)$ is unique because $x + (000\dots0001) = y + (0000\dots0001)$ and $(000\dots00001) + (0000\dots00001) = (000\dots00000)$, and $x = y$ Therefore $f(x)$ is unique. Therefore we have a bijection between $f(x)$ and x which are even and odd weighted values. Thus the set of odd and even nodes are the same size.*

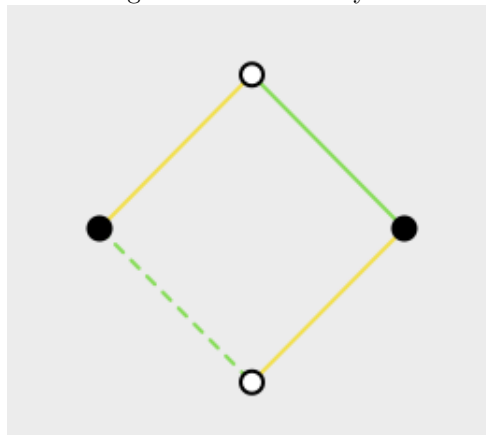
This theorem is actually necessary to show that the nodes adhere to the bipartite, complete, and full, structure of the graph.

Dashedness-We can incorporate dashedness into these Adinkras, however the mathematical

reasons we do so are going to be in a later chapter. For this introduction we should simply consider a cycle of edges around 4 nodes. We define a 4 cycle as traveling by 2 colors, calling these colors z and a . We travel along color z than a than z than a . This will return us to our original node. We must have odd dashedness through the 4 cycle...this means that for every 4 edges that start at node i , and return to node i , we must have an odd amount of dashes.

[h]

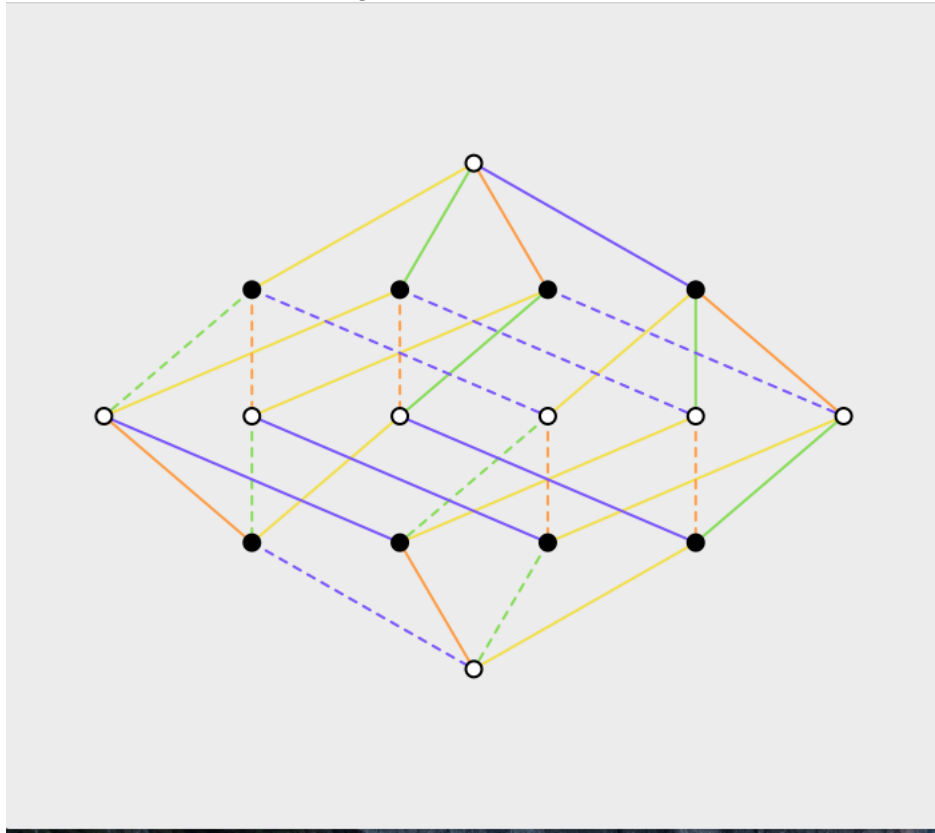
Figure 3.3.1. The 4 Cycle



The Adinkra-We can now look at a full Adinkra. In a moment I will walk through drawing one from scratch, but for now, to define the terms I use, let's look at the 4 dimensional hypercube. Below is a full adinkra. if we look at this figure we can see a number of important relationships. As discussed in the previous writings. First we have a node structure in the binary code. It should be noted that these nodes are at four different heights. (000) is at the lowest height. (100),(010),(001) are all at the second height, and (110)(101)(011) are at the third height, and (111) is at the fourth height. At first glance this figure looks like a cube but it serves us greater usage to look at it as a 2 dimensional picture with 4 different heights. We can also clearly see that the weight of the bits corresponds to the height region of the cube. When we organize adinkras this is the typical method of doing so.

Please examine the color lines from (000) to (100),(010),(001). As we can quickly see there are three lines of color green, red, and orange that extend. Green extends from (000) to (100),

Figure 3.3.2. 4d Adinkra



orange extends from (000) to (010), red extends from (000) to (001). We can also quickly see that regardless of dashing we have three actions corresponding to color.

Remembering that these binary codes are a $Z_2 \oplus Z_2 \oplus Z_2$, a transfer along a green line corresponds to adding (100), a transfer along orange corresponds to adding (010) to any bit code, and a red edge corresponds to add (001) to any bit code. We can use this universally, if we want to get from (010) to (001) we would travel along orange, and then a red line, and this would take us along this path. We can also travel along a red line, and then an orange. One can quickly see that this works by doing the arithmetic, $(010) + (010)(orange) + (001)(red) = (001)$. This works for any transformation we would like to make. There is a lot more nuance to why this is and where these edges come from in the physical realm, but that will be in a separate chapter that deals with the physics of all of this.

From here we have constructed the basic hyper cube. The Adinkra that this paper is most concerned with, and that will be referenced the most is the 4 dimensional Adinkra. The 4 dimensional Adinkra will have a hypercube form that ranges from bitstring (0000)...(1111)

Definition 3.3.2. height and orientation- In Adinkras there exists a way to define a direction to edges and a corresponding height assignment to nodes. although we are not particularly concerned with this in my research, the assignment is a property of Adinkras. When we assign direction to an edge, we define height as a directional edge going from vertex b to vertex a has $hgt(b) = hgt(a) + 1$ [13]

The Height has a meaning in the physics as well. When we have an increase in height, we are making a change in the engineering dimension defined by ∂_τ by differentiation. When we have a reduction in height assignment, we are integrating on the engineering dimension.

4

Quotienting and the Valise

It was shown in *Codes and Supersymmetry in One Dimension* that every connected Adinkra can be obtained by Quotienting the N-cube by a doubly evening code. [4]

Definition 4.0.1. For a group G and a normal subgroup N of G , the quotient group of N in G written G/N , is the set of cosets of N in G . The elements of G/N are written $aN, a \in G$ form a group under the normal operation of the group G on the coefficient a .

Definition 4.0.2. . A n -dimensional chromotopology is a finite connected simple graph A . such that. 1. A is n -regular and bipartite(same number).

Definition 4.0.3. A pretopology is a n -regular finite connected multigraph. A prechromotopology is a generalization of a chromotopology where the corresponding graph can be a pretopology rather than just a topology

[13] “Consider the n -cubical chromotopology Z_c^n . For any linear code $L \subset Z_2^n$, the quotient Z_c^n/L is a \mathbb{Z}_2 -subspace. Using this, we define the map p_L , which sends Z_c^n to the following prechromotopology, which we call the graph quotient (or quotient for short) Z_c^n/L : let the vertices of I_c^n/L be labeled by the equivalence classes of Z_c^n/L and define $p_L(v)$ to be the image of v under the quotient Z_c^n/L . When L is an (n, k) -code, the preimage over every vertex in I_c^n/L contains 2^k vertices, so I_c^n/L has 2^{nk} vertices” [16]

For the 4 cube we have only one doubly even subgroup and that is

$\{(1111)(0000)\} = \langle(1111)\rangle$. Let's now consider the quotient group generated by G/N . This group

is

$\{(0000), (1111)\}, \{(1000), (0111)\}, \{(0100), (1011)\}, \{(0010), (1101)\}$

$\{(0001), (1110)\}, \{(1100), (0011)\}, \{(1010), (0101)\}, \{(1001), (0110)\}$

We form an equivalence relation by setting all nodes in a coset to be equal. By setting these nodes equal we are left with only 8 nodes. The even weight nodes are bosons and odd nodes are fermions.

We have $[0000], [1010], [1100], [1001]$ as a list of bosons which have all been set in an equivalence relation with the other even nodes in Z_2^n , we also have the nodes $[1000], [0100], [0010], [0001]$ are matched with the other odd nodes. This process is ensured by an even code quotienting process.

A valise Adinkra is an Adinkra where all bosons have the same engineering dimension, and all fermions have the same engineering dimension. These valise Adinkras are the main focus of this paper. Through a number of operations, any Adinkra can be put in valise form.

When a valise is created, and we match the nodes as in the example from above, we are physically moving the nodes of the Adinkra, to match each other in the equivalence classes generated by the Quotienting. Zhang gives us a mathematical description of moving these nodes as well. It is entitled the "Hanging Garden Theorem" and is an essential piece of the math of Adinkras. This theorem is given by C. Doran in one of his papers, but is well explained in *Adinkras for Mathematicians* by Zhang.

Fix a chromotopology A . Let $S \subset V(A)$ and $hS : S \rightarrow Z$ satisfy the following properties:

1. hS takes only odd values on bosons and only even values on fermions, or vice-versa
2. For every distinct s_1 and s_2 in S , we have distance defined by $D(s_1, s_2) \geq |hs(s_1) - hs(s_2)|$.

Then, there exists a unique ranking of A , corresponding to the rank function h , such that h agrees with hS on S and A 's set of sinks is exactly S . By symmetry, there also exists a unique ranking of A whose set of sources is exactly S . In other words, any ranking of A is determined by a set of sinks (or sources) and the relative ranks of those sinks/sources. We can think of such a choice as the following: pick some nodes as sinks and pin them at acceptable relative ranks, and let the other nodes naturally hang down. Thus, Theorem 6.2 is also called the Hanging Gardens Theorem" [3]

When we draw the valise we consider the equivalence class generated from the quotient group. When we set all elements of each coset equal to one another, we can match nodes graphically in an Adinkra. When we quotient by a doubly even code we notice that edge color matches up in pairs, and dashing also remains consistent (although this process is far more complicated and not relevant to this paper). This allows us to build an Adinkra valise consistently.

A few things that are different about this valise from the original Adinkra is that we have only two height assignments. These two height assignments are the bottom and top of the valise. The height assignments remain in the same definition from above. The bottom height assignment is the boson nodes, where the top height node are the representations of the fermion nodes.

We can create valise Adinkras for Adinkras with larger susy generators as well. For the 6 susy generator Adinkra example we have 2 different permutation equivalence classes. The G , (quotient group), can be $\langle(110011)\rangle$ and $\langle(111100), (001111)\rangle$. To make this valise, we would go through the same process as with the 4 susy generator example.

4.1 The Adjusted Adjacency Matrix

A large consideration in this paper is the notion of the adjacency matrix. The term "adjacency matrix" in normal graph theory means something different than how I am using the term here.

It is important that I am using notation that I will define, but is different than what is used normally in graph theory.

As far as adjacency matrices go, everything that follows will depend on this concept. The point of the adjacency matrix is to encode the Valise Adinkra as a set of matrices. I will build an adjacency matrix that will first represent an Adinkra valise, and explain how this works.

The two matrices below are the valise matrices representing the chiral multiplet, and the vector multiplet that we derived above. We were able to build 4 matrices L_i , representing the action of each generator of the susy on the bosons in these multiplets. To build a valise from the quotienting in the section above, we start with 4 nodes on the bottom (bosons) which all are in an equivalence class with other bosons from the 4-cube, and the 4 nodes on the top (fermions) which are all in an equivalence class with other fermions for the 4-cube.

Figure 4.1.1. Chiral Multiplet

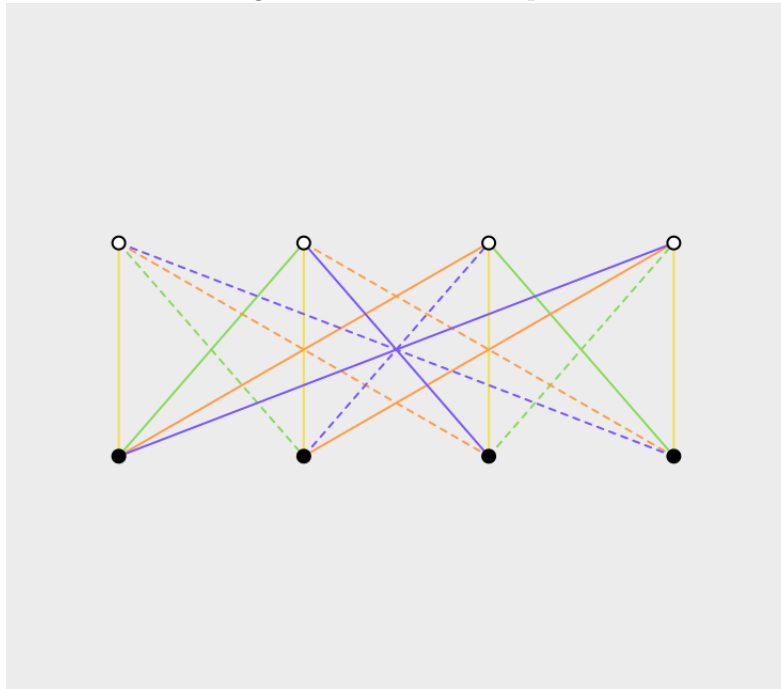
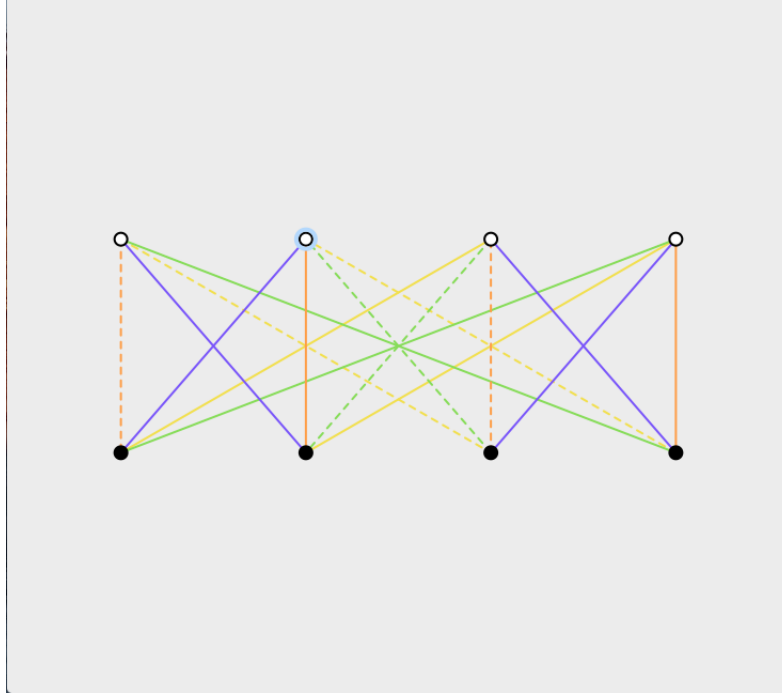


Figure 4.1.2. Vector Multiplet



From here we take each of the matrices generated from one of each of the susy generators. In the case of the Chiral and Vector Multiplets there are 4. This gives the valise the 4 susy transformations on the bosons. Taking the first matrix of the Chiral Multiplet, I_4 . Now we consider each boson reading from left to right, We label these bosons, boson 1, boson 2, boson 3, boson 4. The same labeling is done with the fermions. Each matrix in the Chiral Multiplet has N rows and N columns. Our row 1, column 1 has a 1 entry, for row 1, column 2-4, we have zeroes. When we draw a valise, row i corresponds from the i th boson. We use column j to denote the j th fermion. When we see a 1 in row i and column j , we draw a colored edge from boson i to fermion j . In the matrices we use dashedness to denote a -1 in the matrix, with a dashed line. Each matrix in a multiplet set is associated with its own color.

By doing this for all matrices in a multiplet we can draw a complete valise. These matrices are called the L_i . These L matrices are all associated with the matrix representations of edges from boson to fermion and represent the action of the i th susy generator on the bosons i.e. define a map on the space of the bosons to the space of the fermions. The matrix representation

of edges from fermion to boson are called the R matrices. In the R matrix the fermions are labeled in the rows, and bosons are in the columns. This is equivalent to saying $R = L^T$

Each boson and fermion will have N edges attached, where N is the original number of susy generators of the full Adinkra. These edges will all be different colors, as each susy generator sends every boson to a unique fermion.

We have a few identities...

identity 4.1.1

$$L = R^T$$

Definition 4.1.1. “The transpose A^T of a matrix A can be obtained by reflecting the elements along its main diagonal. Repeating the process on the transposed matrix returns the elements to their original position. In linear algebra, the transpose of a matrix is an operator which flips a matrix over its diagonal, that is it switches the row and column indices of the matrix by producing another matrix denoted as A^T . It is achieved by any one of the following equivalent actions:

reflect A over its main diagonal (which runs from top-left to bottom-right) to obtain A^T ,

write the rows of A as the columns of A^T ,

write the columns of A as the rows of A^T

[15] A transpose flips a matrix over its diagonal, or switches the row and column value of the matrix. Because the L matrix, sends boson(row) to fermion (column). The R matrix sends fermions(row) to bosons (columns).

We denote bosonic nodes as row entries in the adjusted adjacency matrix reading from left to right. The top nodes, the fermions we denote as column entries, reading from left to right. This means that if we have a z -colored line from boson 1 to fermion 3, the L matrix denoting the

action of z-color will have a non-zero entry in row 1 column 3. **identity 4.1.2** $L_i R_i = I$

It is obvious to see this graphically. An L_i matrix denotes an action from a boson to a fermion along a specific colored edge, of which there is only one edge of this color incident to any specific node. The R_i matrix sends a fermion to a boson along this same colored edge. The result will always send a node back to itself. This graphically proves the identity.

4.2 The L matrix

When we consider any of the L matrices in a multiplet we can consider a specific property essential to any of these individual matrices. Let's look at for example a matrix from the Chiral Multiplet.

$$D_a F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Which is one of the matrices in the chiral multiplet example. We can rewrite this matrix as the product of two matrices...

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

In this product we can see that we have a singular permutation matrix, and another matrix that operates on the permutation matrix to negate certain rows. This will be important in later notes.

We can write this in another way, in this first set of matrices that the negation matrix negates the rows aligned with the places where the identity matrix has -1 values.

$$D_a F = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

In this case the negation matrix operates on the columns of the permutation matrix, this is another way to write the singular adjacency matrix as a product of two matrices.

4.3 Valise and Matrix Representation

The valise can be built without the direct process of quotienting. Specifically we can use the Valise to represent the Chiral Multiplet, and Vector Multiplet. These is because the Chiral and Vector Multiplet generate unique adinkras. We have 4 matrices associated with the action of the 4 transformations created by the Multiplets. Like the adjacency matrices are derived from the Visual representation of a valise, we can build a valise from a set of adjacency matrices.

The first is the valise built from the Chiral Multiplet. Remember, the equations have a domain of bosons and a range of fermions. This means that when we build a valise from matrices, we must go from boson to fermion. With this all said, The first representation is the the Chiral Multiplet.

We can see here that the gold lines represent the first matrix $D_a A$, The green line represents $D_a F$, the orange line represents $D_a G$, and the Blue line represents $D_a B$.

We also have a valise representation for the Vector Multiplet that we defined in Chapter 3.

Again we can quickly map out the Adinkra transformations from this valise. The orange edges corresponds to $D_a A_3$, blue corresponds to $D_a A_1$, Yellow corresponds to $D_a d$, Green corresponds to $D_a A_2$,

The abstract point here is that for any matrix set you get that has a symmetry representation, you can make a valise out of any set of matrices, as long as you follow the form set by the adjacency matrices.

4.4 The Node Flip

There are a few values that we are very interested in graphically when we are studying adinkras. One of these is what occurs when we flip all the dashedness from a single boson or fermion in a valise. This means that the dashedness of every edge incident to a node flips. The question is, what happens to the adjacency matrix when we perform this action on a particular valise.

When we flip a node, we are flipping any solid edges to dashed edges incident to a node, and any dashed edges to solid edges. Put simply the action on any edge incident to the flipped node would be to change the dashedness.

In the adjacency matrix representations what occurs is a denoted “1” becomes “-1” and denoting a “-1” to a “1”. The question is generally, what occurs to a specific set of adjacency matrices representing a valise, when a node is ”flipped”. To answer this we must remember that we are working off a specific node, which means that we are negating one entry in each row of an L matrix (assuming we flip a boson). This is because the edges incident to each boson are represented by one row or column in either the L or R matrices. Whichever boson or Fermion we select then, we must simply negate the corresponding row or column of each L matrix. The node flip of boson (i), will affect the i th row of all L matrices. The node flip on a boson will affect the columns of an R matrix.

Figure 4.2 is the example of the Chiral Multiplet with a Node flip done on the second boson. Rewriting the adjacency matrices attached to this valise we get.

$$\text{Gold lines} = L_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \text{ Blue Lines} = L_2 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{Green Lines} = L_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \text{ Orange Lines} = L_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

What we can see here is that every node in row 2 for all L matrices is negated. Row 2 represents every edge incident to our 2nd boson. If we take the transpose of this same operation we get the R matrices. It is obvious that in this situation every entry in column 2 has been negated. Is there a way to represent this action with matrix notation.

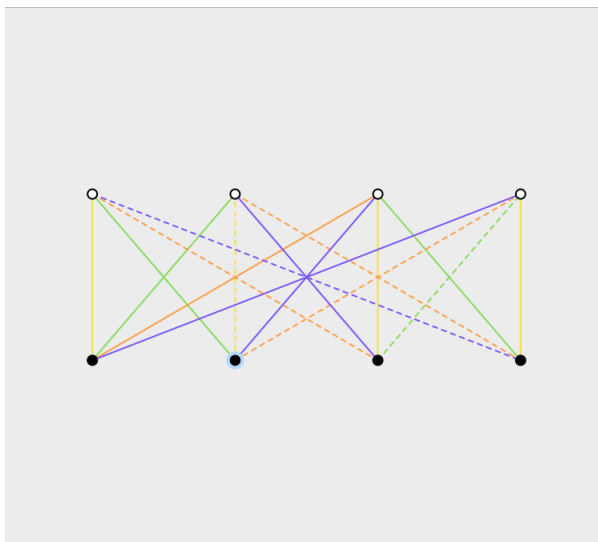


Figure 4.4.1. Chiral Multiplet: boson 2 node Flip

Definition 4.4.1. To represent a node flip on the set of L_i or R_i , we can multiply a matrix N to our whole set of matrices. N is an identity matrix with a negation in row i . Negating boson i is represented by NL_i , and R_iN . If we negate fermion i we can represent this as L_iN and NR_i .

We can adjust the second piece of this definition. We can rewrite L_iN as $\tilde{N}L_i$ where the \tilde{N} has a negative in column j , where boson i acts on L_i . (If node flip is fermion 2, and this L_i has "1" in row 2 column 3, N has -1 in row 2, and \tilde{N} has -1 in row 3) To show that this works in a

general case, we can write an L matrix will be the general form.

$$\begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

The $N * L$ product is then

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ -e & -f & -g & -h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

4.5 The Node Swap

Another value that we are very interested in is what occurs when we flip the i th and j th Bosonic node in the valise Adinkra. Below is a picture of the node flip of the 2nd and 3rd node of chiral multiplet valise.

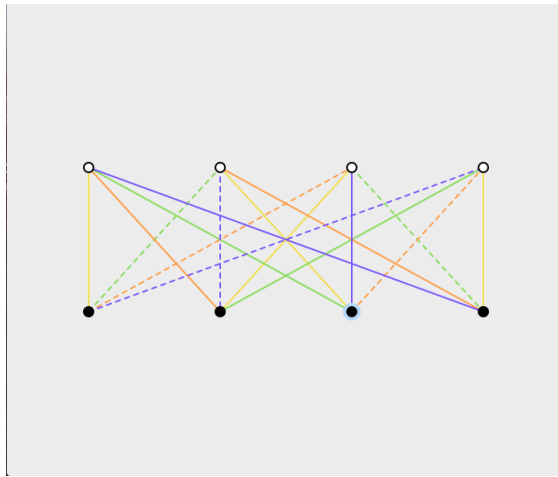


Figure 4.5.1. Chiral Multiplet: boson 2 and 3 swapped

We want to define what occurs when we flip the two nodes in the adjacency matrices.

Definition 4.5.1. flipping two bosons, i and j , can be represented by multiplying matrix PL_i , and R_iP , where P is a permutation matrix with an i and j row swap.

When we swap the i th and j th nodes of a boson, we multiply the adjacency matrix, by a permutation matrix, that swaps the identity matrix i th and j th row, $P*L$. We can do the same when we flip fermions, however, we must multiply by the permutation matrix, $L*P$. This P is a permutation matrix, and the L is the Adjacency matrix.

These matrices act in the same way that the negation matrices act above.

Consider i th node and j th node. i th node works in row a . j th node works in row b . If we swap these two nodes, we are swapping the rows that these nodes operate on. A th node now works in row b and B th node works in row a . This by inspection is obvious.

We can observe this in general, as it obeys this law. For any $n * n$ matrix, we can simply swap rows of the identity matrix. For instance, to permute (13)(24)(31)(42)...

We swap rows 1 and 3 and 2 and 4 of the L matrix, if the L matrix is

$$\begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix}$$

This transformation would look like this on the matrix,

$$\begin{bmatrix} i & j & k & l \\ m & n & o & p \\ a & b & c & d \\ e & f & g & h \end{bmatrix}$$

We can quickly show that the permutation matrix

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} * \begin{bmatrix} a & b & c & d \\ e & f & g & h \\ i & j & k & l \\ m & n & o & p \end{bmatrix} = \begin{bmatrix} i & j & k & l \\ m & n & o & p \\ a & b & c & d \\ e & f & g & h \end{bmatrix}$$

We can see here the general form of the permutation matrix operating on the L matrices.

4.6 Valise Matching

There is a huge number of potential multiplets that can be built from symmetry and supersymmetry equations in many dimensions. When two of these give matrix representations, it is interesting too physicists and mathematicians alike too determine whether under a certain set of rules, one can be made into the other. These rules are as following.

- 1. Any Node Flip, boson and fermion**
- 2. Any Node Swap, boson and fermion**
- 3. Negating the value of any one matrix**

In the case of the Chiral and Vector multiplet, here is how we can make the Vector multiplet into the Chiral multiplet under these set of rules.

The Chiral Multiplet, as defined under a basis change from our chapter in physics. We will

rename the matrices in this particular section C_i $C_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$,

$$C_2 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$C_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$C_4 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

The other set is the vector multiplet, also under a change of basis from the physics. We will

name these W_i .

$$W_1 = I_4$$

$$W_2 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$W_3 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

I want to try to transform our second set of matrices into our first set, so that I can set make our first matrices our second. I can flip all edges, incident to a specific node. Exchange any two nodes, and flip all edges of any one type. These transformations convert to the matrices. Flipping all nodes incident to one edge, is the same as negating a row or column, of each matrix. The node exchange, in matrix form is the same as changing the rows. Finally, the flip of one edge, swaps the sign of every unit in the matrix.

Here's How...

Flip the 2nd and 4th matrices of the vector multiplet

$$W_2 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

Flip the 1st matrix of the chiral multiplet.

Now swap nodes 1-3 and 2-4 of the vector multiplet

$$W_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$W_3 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

Now we negate our third row...or flip the edges incident to our third node.

$$W_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$W_3 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

And finally we flip the second column, which is essentially swapping the nodes incident to the second fermion.

$$W_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$W_3 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

These operations make these multiplets equivalent to each other. I

believe there is a general way to make any of these matrices, into each other. I will use my method on the next set of matrices and explain the idea.

It should be noted that this solution is not unique, the difference between solving out these matrices in order to get a new specific set, is solved in the same general method.

5

My Results, Specific Values of Adinkras

5.1 $\frac{1}{2}(L_i R_j - L_j R_i)$

Adinkras are values that take a 4 dimensional supersymmetry transformation and parameterize the value down to 1 dimension. We demonstrated this in chapter 2. The problem of transforming a 4 dimensional object down to 1 dimension is fairly intuitive, and is neither a new nor difficult one to solve with proper tools. The problem of restoring 1 dimension to 4 is much more difficult. Part of what Physicists and Mathematicians hope to gain from the use of adinkras is an understanding of how various transformations in 1 dimensional supersymmetry affects supersymmetry in 4 dimensions. The task of my senior project has been to evaluate one of these values, that was considered to be invariant under certain transformations by a number of physicists. It was however not.

When we evaluate $\frac{1}{2}(L_i R_j - L_j R_i)$ it is helpful to consider it in the case where $\mathcal{N} = 4$, (the case of 4 susy generators), as discussed before. This case will give us a general understanding of what I am trying to prove in the future. To understand the particulars of the problem, there are a few properties that will help the reader understand the math I have done at a later point.

5.2 Possible Permutations

In the $\mathcal{N} = 4, K = 1$ case there are only a few possible structural forms of the matrix. As discussed in section 4.2, any of the L matrices can be represented as the product of a Permutation matrix and a Row or Column Negation Matrix.

The first question in evaluating the $\frac{1}{2}(L_i R_j - L_j R_i)$ is, what possible permutation forms can exist. The Negation matrix will not have any part in this, so the question is $\frac{1}{2}(|L_i R_j - L_j R_i|)$ has which values?

It turns out that in the case of the basis selected, for the vector and chiral multiplet, can only have the following forms, $|L_i|$. This is not the only set of adjusted adjacency matrices we can have in our multiplets, but what will be consistent is that for any basis there will be only 4 forms, and they will all be permutations of the following.

1. I_4

$$2. \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}, 3. \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, 4. \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

We can rewrite these matrices in cyclic notation:

$$1.(1)(2)(3)(4)$$

$$2.(12)(34)$$

$$3.(14)(23)$$

$$4.(13)(24)$$

These are all of the possibilities.

This means that we can quickly and simply evaluate $|L_i R_j|$ as a product of any two adjacency matrices in a simple permutation form.

5.3 The Permutation Matrix Group

I will now define very specific group that I will call the Permutation Matrix Group. This group is made up of the identity matrix with all possible permuted rows and columns. The L_i matrices are a subgroup of this group.

1. The inverse of any permutation matrix is itself
2. The group holds under multiplication.
3. I is in this group

Theorem 5.3.1. *Let $Q_n, P_n \in$ Permutation group*

Let $Z_n = Q_n P_n$ Since the group is closed under multiplication, Z_n is a permutation matrix.

$$I = I$$

$$I = Z_n Z_n \text{ by 1.}$$

$$I = (Q_n P_n)(Q_n P_n)$$

$$P_n = (Q_n P_n Q_n)$$

$$P_n Q_n = Q_n P_n$$

Then this group is commutative under multiplication.

Recognizing that the permutations from our L matrices are group elements themselves, independently (this can be shown under simple computation), is important in understanding the deeper aspects of this paper. Although this does not have direct implications in my work, it

shows some of the thought process of working with multiplicative combinations of L and R matrices.

5.4 The Permutation Form $L_i R_j$

We want to find the general permutation form of the product $\frac{1}{2}(L_i R_j - L_j R_i)$ in a simple generalized way.

1) $L_i R_i = I$. Because $L_i = R_i^{-1}$

Therefore $L_i R_i - L_i R_i = I - I = 0$

2) We can rewrite the matrices in cycle notation as $(ab)(ij)$. In the case of the identity matrix, $a = b$ and $i = j$. However, it will serve us better to simply write I as $(a)(b)(i)(j)$. All of these matrices are in this form.

3) The Garden Algebra that acts on the L matrices guarantees that $L_i R_j = -L_j R_i$

This will be the most important result.

The Garden Algebra states that $L_i L_j^T + L_j L_i^T = 2\delta_{ij}$. The Kronecker delta is 1 when $i = j$.

For all other cases it is 0. This means that for the case where $i \neq j$ we get $L_i L_j^T = -L_j L_i^T$. This identity will be key.

There are a few important properties of this identity. Remember we can write any of the special adjacency matrices as the product of two matrices, a permutation matrix and a negation matrix.

We also know that the product of any two adjacency matrices here is another matrix with the same properties. Namely, that the rows and columns are linearly independent, and the entries other than 0 are 1, or -1. Let's call the product of $L_i L_j^T = S_i$. We can rewrite S_i as a permutation matrix T_i and a negation matrix N_i :

$S_i = -L_j L_i^T$ From the Garden Algebra, we can immediatly see that $-S_i = L_j L_i^T$

5.5 The permutation of $\frac{1}{2}(L_i R_j - L_j R_i)$

I want to consider the relationship between $L_i R_j - L_j R_i$ and what occurs when we do both a flip on any particular node, and a swap on any two nodes, i and j.

For simplicity sake let's define three values, $L_i R_j - L_j R_i = \nu$

Let's then define the same value, with a node flip. We have defined the operation of a node flip with adjacency matrices above. Let's call a node flip on i, κ , We wish to evaluate $\kappa L_i \kappa R_j - \kappa L_j \kappa R_i$

We need to be careful when considering how to evaluate κ on the R adjacency matrices where we need to transpose. When we do a node flip on a boson, we are negating the columns of the R adjacency matrices as defined in chapter 4. I now propose that we denote N_i as the negation matrix from chapter 4, flipping boson i. This means we can write the second value as such.

$$N_i L_i R_j N_i - N_i L_j R_i N_i$$

Flipping fermion i yields

$$L_i N_i N_i R_j - L_j N_i N_i R_i.$$

This is in accordance with the matrix properties defined in section 4.4.

Finally we want to consider the third value, and this is the most complicated and most interesting. We know that a permutation matrix multiplied by a adjacency matrix, represents a node flip. We need to be very very careful with these values. let's call this swap ξ . Like above, we need to compute these values after a node flip. The same action must be considered from the nodes above.

Considering the action of ξ on the special value I am examining in this chapter, define a permutation matrix representing the flip of nodes i and j as P_{ij} . From section 4.5, the matrix representation $(L_i R_j - L_j R_i)$, under the action of a node swap of bosons i and j can be written as, $P_{ij} L_i R_j P_{ij} - P_{ij} L_j R_i P_{ij}$. On the action of swapping two fermions i and j , we can write, $L_i P_{ij} P_{ij} R_j - L_j P_{ij} P_{ij} R_i$.

5.6 First Big Proof

The susy models based on the dirac matrices can always be written as a product of two,2-cycles in permutation notation. First simply choose your cycles. These can represent the permutation of L_i in the basis, provided that L_i is not the identity. I wish to evaluate $L_i R_j$. However if I cannot represent L_I in this permutation form, then I can evaluate $L_j R_i$, since it is the negation of $L_i R_j$ and invariant under the absolute value.

The form of the V matrix given by $\frac{1}{2}(L_i R_j - L_j R_i)$ will be given by the result of the first $L_i(L_j^T)$. This follows, because we know from the Garden Algebra that $(L_i)(L_j)^{(T)} = -(L_j)(L_i)^T$ since form only refers to the absolute values. If we let our first $L_i = (ai)(bj)$, we know that L_j can equal $(ai)(bj)$, $(ab)(ij)$, or, $(aj)(bi)$. These are all of our unique permutations multiplets, other than the identity.

We can build a small set of equivalence relations from these values

There are four cases:

1 The L_i and L_j^T are the same matrix.

$$(ai)(bj)$$

$$(ai)(bj)(ai)(bj) = I = V_1$$

Now to flip the I and J node

$$(ij)(ai)(bj)(ai)(bj)(ij)$$

$$(ibja)(ajbi) = I = \widetilde{V}_1$$

Therefore $V_1 = \widetilde{V}_1$ **2** Let us now consider the permutation where the second matrix is $(ab)(ij)$

Then

$$(ai)(bj)(ab)(ij) = (aj)(ib) = V_2$$

And with the swap

$$(ij)(ai)(bj) = (ibja)$$

$$(ab)(ij)(ij) = (ab)(i)(j)$$

$$(ibja)(ab) = (ia)(bj) = \widetilde{V}_2$$

$$V_2 = (ji)(ab)\widetilde{V}_2$$

Since $j \neq i$ and $a \neq b$ $V_2 \neq \widetilde{V}_2$ **3**

The third matrix option of the second matrix $(aj)(bi)$

$$(ai)(bj)(ab)(ij)$$

$$(aj)(ib) = V_3$$

With the i and j swap,

$$(ij)(ai)(bj) = (ibja)$$

$$(aj)(bi)(ij) = (aibj)$$

$$(ibja)(aibj) = (ij)(ab) = \widetilde{V}_3$$

We can see that $V_3 = \widetilde{V}_3$ **4**

and finally, the second matrix is an identity

$$(ai)(bj) = (ai)(bj) = V_4$$

and with a node swap

$$(ij)(ai)(bj) = (ibja)$$

$$I(ij) = (ij)$$

$$\begin{aligned}
(ibja)(ij) &= (ib)(ja) = \widetilde{V}_4 \\
V_4 &= (ab)(ij)\widetilde{V}_4 \\
V_4 &\neq \widetilde{V}_4
\end{aligned}$$

Therefore the set of results can be displayed as such

$$\{V_1, V_2, V_3, V_4\} = \{\widetilde{V}_1, (ia)(bj)\widetilde{V}_2, \widetilde{V}_3, (ab)(ij)\widetilde{V}_4\}$$

The second set can be written as $\{\widetilde{V}_1, L_{j(1)}\widetilde{V}_2, \widetilde{V}_3, L_{j(2)}\widetilde{V}_4\}$

In the case where the first matrix L_i is an identity, we can use our rules to evaluate $L_j L_i^T$ instead.

Remember, because of the Garden Algebra, $L_i J_j^T - L_j L_i^T = 0$ if $i = j$

5.7 The Bigger Proof

Given any set of V_i , as the result of the computation $\frac{1}{2}(L_i R_j - L_j R_i) = V_i$ for ANY set of L_i following the physics defined in the paper, and the Garden Algebra as defined in this section, we can determine what will occur to V_i under the case of a node flip, or node exchange of bosons alone:

$$\frac{1}{2}(L_i L_j^T - L_j L_i^T) = V_i$$

$$\frac{1}{2}(S_i - (-S_i)) = V_i$$

$$S_i = V_i$$

Under the case of a node exchange, (ij) we have shown we can represent this node exchange as a permutation matrix which we will call P_{ij}

We have also shown how this permutation will act on the values.

In the case of a node exchange of bosons,

$$\begin{aligned}\frac{1}{2}((P_{ij})L_iL_j^T(P_{ij}) - (P_{ij})L_jL_i^T(P_{ij})) &= \tilde{V}_i \\ \frac{1}{2}((P_{ij})(S_i)(P_{ij}) - (P_{ij})(-S_i)(P_{ij})) &= \tilde{V}_i \\ \frac{1}{2}((P_{ij})(S_i)(P_{ij}) - (P_{ij})(-1)(S_i)(P_{ij})) &= \tilde{V}_i \\ \frac{1}{2}((P_{ij})(S_i)(P_{ij}) - (-1)(P_{ij})(S_i)(P_{ij})) &= \tilde{V}_i \\ (P_{ij})(S_i)(P_{ij}) &= \tilde{V}_i\end{aligned}$$

And so we have a general formula for a node exchange on any nodes in the Valise Adinkra.

Now for the node flip. Again we know that we can represent a node flip by multiplying the values by a negation matrix N with a negative in the row or column corresponding to the flipped node.

That looks exactly the same

We can simply replace the P_{ij} matrix with N_i where i is the flipped node

For a node flip on any boson. $\frac{1}{2}(N_i)(S_i)(N_i) - (N_i)(-S_i)(N_i) = V_i$

$$(N_i)(S_i)(N_i) = V_i$$

by the exact same calculation.

5.8 Fermion Invariance

Operating on $\frac{1}{2}(L_iR_j - L_jR_i)$ by a permutation of two fermions was perviously shown to be $\frac{1}{2}(L_iP_{ij}P_{ij}R_j - L_jP_{ij}P_{ij}R_i)$.

We know that the product of the same two permutations is the identity, as they are a member of the permutation group from Section 5.2.

$$P_{ij}P_{ij} = I.$$

Therefore swapping the nodes of two fermions yields, $\frac{1}{2}(L_iR_j - L_jR_i)$. Therefore swapping two fermions is invariant on the value.

We also know that the product of the same two negation matrices is the identity. This means that the negation of a fermion in the Adinkra yields, $\frac{1}{2}(L_iN_iN_iR_j - L_jN_iN_iR_i) = L_iR_j - L_jR_i$. This is equivalent to negating a fermion is invariant in our value.

These are essential results, as they are far different than the process of swapping or negating a boson.

5.9 Future Work

I believe that this general proof has not yet been done in the mathematical fields of Adinkras. I have no reason to not believe it. This is therefore the culmination of my senior project. There is a lot more to do, and I would like to have some of my work published if I can, and hopefully, after graduation I can take steps to do so.

There are certainly more questions, some of which I believe I have answers too, but have not been able to prove solidly without more work. One interesting problem is, can we do some negation on these values and some permutation, and get the same thing? Is this even possible? If so, under what rules. This is another challenging problem I will hopefully attempt in the future. There are many other challenges in the work that I am doing, and there is much more to know. Trying to find large solutions for other values than the one I explored in my project is essential to the process of moving from 1 dimension back to 4. In the future years, these results will come out quickly.

My hope is that I will be involved in this process, as I continue my education.

Thankyou

Bibliography

- [1] Mathew Calkins, D. E. A. Gates, S. James Gates, and Brian McPeak. Is it possible to embed a $4D, \mathcal{N} = 4$ supersymmetric vector multiplet within a completely off-shell adinkra hologram? *JHEP*, 05:057, 2014.
- [2] Isaac Chappell, II, S. James Gates, and T. Hbsch. Adinkra (in)equivalence from Coxeter group representations: A case study. *Int. J. Mod. Phys.*, A29(06):1450029, 2014.
- [3] C. F. Doran, M. G. Faux, S. J. Gates, Jr., T. Hubsch, K. M. Iga, and G. D. Landweber. On graph-theoretic identifications of Adinkras, supersymmetry representations and superfields. *Int. J. Mod. Phys.*, A22:869–930, 2007.
- [4] C. F. Doran, M. G. Faux, S. J. Gates, Jr., T. Hubsch, K. M. Iga, G. D. Landweber, and R. L. Miller. Codes and Supersymmetry in One Dimension. *Adv. Theor. Math. Phys.*, 15(6):1909–1970, 2011.
- [5] Charles Doran, Kevin Iga, Jordan Kostiuk, Greg Landweber, and Stefan Mendez-Diez. Geometrization of N -extended 1-dimensional supersymmetry algebras, I. *Adv. Theor. Math. Phys.*, 19:1043–1113, 2015.

- [6] Charles Doran, Kevin Iga, Jordan Kostiuk, and Stefan Mndez-Diez. Geometrization of N -Extended 1-Dimensional Supersymmetry Algebras II. 2016.
- [7] Michael Faux and S. J. Gates, Jr. Adinkras: A Graphical technology for supersymmetric representation theory. *Phys. Rev.*, D71:065002, 2005.
- [8] Isaac Friend, Jordan Kostiuk, and Yan X. Zhang. Enumerative Gadget Phenomena for $(4, 1)$ -Adinkras. 2018.
- [9] S. J. Gates, Jr., J. Gonzales, Boanne MacGregor, J. Parker, Ruben Polo-Sherk, V. G. J. Rodgers, and Luke Wassink. 4D, $N = 1$ Supersymmetry Genomics (I). *JHEP*, 12:008, 2009.
- [10] S. James Gates, Kevin Iga, Lucas Kang, Vadim Korotkikh, and Kory Stiffler. Generating all 36,864 Four-Color Adinkras via Signed Permutations and Organizing into ℓ - and $\tilde{\ell}$ -Equivalence Classes. 2017.
- [11] Greg Landwebber. Adinkramat. 2011.
- [12] H. Miyazawa. Baryon Number Changing Currents. *Progress of Theoretical Physics*, 36:1266–1276, December 1966.
- [13] Sylvia Naples. Classification of Adinkra Graphs. pages 10–25, 2009.
- [14] Madeleine Weinstein. Adinkras and arithmetical graphs. 2016.
- [15] Wikipedia contributors. Transpose — Wikipedia, the free encyclopedia, 2018. [Online; accessed 1-December-2018].
- [16] Yan X Zhang. Adinkras for Mathematicians. *Trans. Am. Math. Soc.*, 366(6):3325–3355, 2014.