GRÖBNER BASES AND SUDOKU PUZZLES

COURTNEY THOMAS

1. Shidoku and Sudoku Puzzles

In this paper, we will discuss and build on the ideas in the article, "Gröbner Basis Representations of Sudoku," by Elizabeth Arnold, Stephen Lucas, and Laura Taalman. We will first develop three ways of representing a Shidoku board as a system of polynomial equations given its constraints. We will then explain how a Gröbner basis of these systems of equations can be used to count the number of Shidoku boards.

Using language from the article [ALT10], we have the following definitions. A *Sudoku board* is a 9×9 grid where the 81 cells are filled with the numbers 1-9 such that each row, column, or 3×3 block contains no repeated entries. A *Sudoku puzzle* is a subset of a Sudoku board that uniquely determines the rest of the solution to the board. A *Shidoku board* is similar to a Sudoku board, but is a 4×4 grid where the 16 cells are filled with the numbers 1-4 such that each row, column, or 2×2 block contains no repeated entries. Thus, a *Shidoku puzzle* is a subset of a Shidoku board that uniquely determines the rest of the solution to the board.

Define a cell to be each individual square. Define a region to be either a row, column, or designated 2×2 square of cells. When we are talking about a particular cell, we will call it by the name $w_{i,j}$. That cell will lie in the i^{th} row and the j^{th} column of the Shidoku board, as shown below.

$w_{1,1}$	$w_{1,2}$	$w_{1,3}$	$w_{1,4}$
$w_{2,1}$	$w_{2,2}$	$w_{2,3}$	$w_{2,4}$
$w_{3,1}$	$w_{3,2}$	$w_{3,3}$	$w_{3,4}$
$w_{4,1}$	$w_{4,2}$	$w_{4,3}$	$w_{4,4}$

Date: May 11, 2014.

2. Polynomial Representations of Shidoku and Sudoku Puzzles

In this section, we will discuss three different polynomial representations of Shidoku: the Sum-Product Shidoku system, the Roots of Unity Shidoku system, and the Boolean Shidoku system.

2.1. **Sum-Product.** To find the Sum-Product system of equations, we seek a set of equations that represent the constraints of a Shidoku board. Using the knowledge that each of the 16 cells may only take on the value of 1, 2, 3, or 4, and the knowledge that each region (as defined above) of 4 cells may only contain the numbers 1, 2, 3, and 4 exactly once, we can derive a system of equations as follows. This system relies on the knowledge that the only way that the numbers $1-4$ can be combined such that they add to 10 and multiply to 24 is if each number is used exactly once. There are four other ways that the numbers 1, 2, 3, and 4 can be combined to add to 10 (namely $\{1,1,4,4\},\{1,3,3,3\},\{2,2,2,4\}$, and $\{2,2,3,3\}$), however, none of these multiply to 24.

First, we will think of each of the 16 cells individually. We know that $w_{1,1},..,w_{4,4}$ can only take on a value of 1, 2, 3, or 4 per our definition of Shidoku puzzle. Thus, we have 16 equations of the form

$$(w-1)(w-2)(w-3)(w-4)=0,$$

where w is replaced by each of the variables $w_{1,1},...,w_{4,4}$.

Now, we suppose that the set $\{w,x,y,z\}$ is a set of four cells that make up any region of the Shidoku board. Since in each region the numbers 1, 2, 3, and 4 are used exactly once, we know the sum of the cells in each region must sum to $1+2+3+4=10$. So, for each of the 12 regions (4 rows, 4 columns, and 4 2×2 squares) containing some $\{w,x,y,z\}$, we know that

$$w+x+y+z-10=0$$

where $w,x,y,$ and z are replaced by any 4 of the variables $w_{1,1},...,w_{4,4}$ that lie within the same region.

Lastly, we must consider the same set $\{w,x,y,z\}$ that makes up any region of the Shidoku board. Since the numbers 1, 2, 3, and 4 are used exactly once in each region, we know that the product of the cells in each region must multiply to $1\cdot 2\cdot 3\cdot 4=24$. So, for each of the 12 regions, we know that

$$wxyz-24=0$$

where $w,x,y,$ and z are replaced by any 4 of the variables $w_{1,1},...,w_{4,4}$ that lie within the same region.

Thus, these 40 equations give us the Sum-Product Shidoku system. This means we could mathematically represent any Shidoku puzzle by adding more equations to specify any known cell values [ALT10]. For example, if we were to use the 40 polynomial equations that we just found, in addition to the following 6 equations: $w_{1,4} = 4, w_{2,1} = 4, w_{2,3} = 2, w_{3,2} = 3, w_{3,4} = 1, w_{4,1} = 1$, we would be representing the Shidoku puzzle below.

			4
4		2	
	3		1
1			

The solution to a system of equations representing a Shidoku puzzle is the solution to a puzzle. For instance, when you solve for $w_{1,1}$ in your system of equations, you should obtain a value of 1, 2, 3, or 4 that would go in the first row and the first column of the puzzle.

2.2. **Roots of Unity.** Next, we will find an entirely different set of equations, which we will call the Roots of Unity system. To find this set of equations that accurately represents the constraints of a Shidoku board, we will use the fact that each of the values 1, -1, i, and $-i$ are fourth roots of unity (meaning that each value to the fourth power is equal to 1). Thus we will use these values instead of the 1, 2, 3, and 4 that we used in the development of the Sum-Product system.

To find the Roots of Unity system, we must remember that the specific symbols used in Shidoku have no effect on the rules or outcome of the board. With this fact in mind, we will replace the symbols 1, 2, 3, and 4 with 1,-1, i, and $-i$. Now, as before, we will start by looking at an equation for each cell individually. Since each of these values is a fourth root of unity, we can easily see that for each of the 16 cells

$$w^4 - 1 = 0$$

where w is replaced by each of the variables $w_{1,1}, ..., w_{4,4}$.

Now, we will consider any two cells w and x that lie in the same region of the board. By the first set of equations, we know that $w^4 - 1 = 0$ and that $x^4 - 1 = 0$. Thus, we know that $w^4 - x^4 = 0$. If we factor this difference of two squares, we get $(w^2 - x^2)(w^2 + x^2) = (w - x)(w + x)(w^2 + x^2) = 0$. Since w and x lie in the same region of the board, we know that they must have different values. So, if $w \neq x$, then $w - x \neq 0$. So, we get that for any w and x in the same region,

$$(w + x)(w^2 + x^2) = 0$$

where w and x are replaced by any 2 of the variables $w_{1,1}, ..., w_{4,4}$that lie in the same region of the Shidoku board.

So, how many of these equations are there? Well for rows, we know that there are $\binom{4}{2}$ possible w and x combinations. So, we have $\binom{4}{2} = \frac{4!}{2! \cdot 2!} = \frac{4 \cdot 3}{2} = 6$ combinations for each of the 4 rows. This combination also holds true for columns, so we have 6 combinations for each of the 4 columns. For each 2×2 block, we have $\binom{4}{2}$ combinations, but we must account for the two rows and two columns within the 2×2 block that we have already accounted for. This gives us $6 - 2 - 2 = 2$ combinations for each of the four 2×2 blocks. Together, that gives us$(6)(4) + (6)(4) + (2)(4) = 24 + 24 + 8 = 56$ equations of this type.

Together with the above 16 equations, we have 72 polynomial equations that represent a Shidoku puzzle using the Roots of Unity system [ALT10].

2.3. **Boolean.** The Boolean system requires us to introduce more variables, but will ultimately assist us in simplifying matters. We will introduce w_1, w_2, w_3, and w_4 for each cell on the Shidoku board. Note that this brings us from 16 to 64 variables. In this system, we will set $w_k = 1$ when the cell w takes on the value k, and $w_k = 0$ when w takes on any of the other three values. By doing this, we get 64 equations (4 for each cell) such that

$$w_k(w_k - 1) = 0.$$

While this may seem overwhelming, we must remember one important fact about Boolean systems. Since cell values are only able to equal either zero or 1, we know that no matter what value a particular variable holds, $w^2 = w$. Thus, during computations, any power of any w_k may simply be replaced by w_k.

We also know that each cell on the board may only hold one value at a time. Thus, only one of our Boolean variables for each cell (w_1, w_2, w_3, w_4) may equal 1, while the rest will be equal to zero. So, we have 16 polynomial equations (one for each cell) of the form

$$w_1 + w_2 + w_3 + w_4 = 1$$

where w is replaced by each of the values $w_{1,1}, ..., w_{4,4}$.

Lastly, we must find some equation to represent the fact that any two cells w and x that lie in the same region of the board cannot hold the same value. So, for any value of k, either w_k or x_k must be 0. Thus, we have 56 equations such that

$$w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 = 0$$

where w and x are replaced by any 2 of the variables $w_{1,1}, ..., w_{4,4}$ that lie in the same region of the Shidoku board.

These 136 equations together make up the Boolean Shidoku system. Although these large systems of polynomials may seem cumbersome, we will use Gröbner bases to handle them [ALT10].

3. Gröbner Bases

Informally, "a Gröbner basis for a system of polynomials is a new system of polynomials with the same solutions as the original, but which is easier to solve and often has additional 'nice' properties [ALT10]." In order to precisely describe a Gröbner basis, there are a few terms from abstract algebra that will be useful.

Definition 1. A *ring* is a set together with two binary operations (denoted $+$ and $\cdot$) such that R is a commutative group under addition, multiplication in R is associative, and the left and right distributive law in R holds [Fra03]. A *polynomial ring* is a set $R[x_1, ..., x_n]$ of all polynomials in the variables $x_1, ..., x_n$ so that the coefficients are in the ring R [Fra03].

Definition 2. An *ideal* of a ring R is an additive subgroup N of the ring R such that for all $a, b \in R$, $aN \subseteq N$ and $Nb \subseteq N$ [Fra03].

Definition 3. An ideal generated by polynomials $f_1, ..., f_s$, is denoted

$$\langle f_1, ..., f_s \rangle = \left\{ \sum_{i=1}^{s} u_i f_i | u_i \in k[x_1, ..., x_n], i = 1, ..., s \right\}$$

where k is a field [AL94].

So, given a system of polynomials, we are able to look at the ideal that these polynomials generate [ALT10]. We will apply this to our systems of polynomials we obtain from the polynomial representations of Shidoku boards.

Definition 4. We define the *lexicographical term ordering* (or *lex*) with $x_1 > x_2 > \cdots > x_n$ as follows: For

$$\alpha = (\alpha_1, ..., \alpha_n), \beta = (\beta_1, ..., \beta_n) \in \mathbb{N}^n$$

we define

$$x^\alpha < x^\beta$$

if and only if the first coordinates α_i and β_i in α and β from the left, which are different, satisfy $\alpha_i < \beta_i$.

So, in the case of two variables x_1 and x_2, we have

$$1 < x_2 < x_2^2 < x_2^3 < \cdots < x_1 < x_2x_1 < x_2^2x_1 < \cdots < x_1^2 < \cdots$$

[AL94]. We will use lexicographical term ordering (in reverse order) for use in establishing what is said to be the *leading term* of a polynomial. By reverse *lex*, we mean that we will be using the ordering $w_{1,1} < w_{1,2} < ... < w_{4,4}$. For us, lexicographical term ordering will allow us to find an ordered basis so that the first polynomial contains only the variable $w_{1,1}$, and each of the following polynomials will contain only the variables used in previous polynomials in addition to one new variable. This will allow us to make certain assumptions that will make it easier to use the basis for counting purposes.

Definition 5. The *leading term* of a polynomial f is the "largest" term of the polynomial as established by lexicographical term ordering (see Defininiton 4). The leading term can be denoted as $\mathrm{lt}(f)$ [ALT10].

The leading term is comprised of both the *leading coefficient* $\mathrm{lc}(f)$ and the *leading power product* $\mathrm{lp}(f)$, such that $\mathrm{lt}(f)$=$\mathrm{lc}(f)\mathrm{lp}(f)$. For example, given the polynomial $a^4 - 10a^3 + 35a^2 - 50a + 24$, we can establish a^4 to be the leading term, since by Definition 4, it is the "largest" term. It is comprised of the leading coefficient, 1, and the leading power product, a^4. We use this definition of leading term in order to define the *leading term ideal* of a set.

Definition 6. The *leading term ideal* of a set S is the ideal $\mathrm{Lt}(S)$ generated by the leading terms of the polynomials in S, where $\mathrm{Lt}(S)$ =$\langle lt(f) | f \in S\rangle$ [ALT10]. For example, given a set of polynomials $S = \{x, x+1\}$, the $\mathrm{Lt}(S)$= $\langle x\rangle$.

Theorem 7. *(Hilbert Basis Theorem) In the ring $k[x_1, ..., x_n]$, if I is any ideal of $k[x_1, ..., x_n]$, then there exist polynomials $f_1, ..., f_s \in k[x_1, ..., x_n]$ such that $I = \langle f_1, ..., f_s\rangle$*[AL94].

Definition 8. A set of non-zero polynomials G in the ideal I is called a *Gröbner basis* for I if and only if the leading term ideal of G is is equal to the leading term ideal of the ideal I generated by G, or $\mathrm{Lt}(G)$=$\mathrm{Lt}(\langle G\rangle)$ [ALT10].

Recall the set from Definition 6, $S = \{x, x+1\}$. We said that the $\mathrm{Lt}(S)$= $\langle x\rangle$. However, the $\mathrm{Lt}(\langle S\rangle)$= $\langle 1\rangle$. So, S is not a Gröbner basis for the ideal $\langle S\rangle$.

In order to compute Gröbner bases when dealing with large systems of polynomials, we will need to use a computer program. We know it is possible to obtain such a basis because of a theorem by Buchberger.

Theorem 9. *(Buchberger's Algorithm) Every ideal $I = (f_{1,\dots,}f_s)$ in $k[X]$ has a Gröbner basis which can be computed by an algorithm* [Rot06].

Definition 10. Let $0 \neq f, g \in k[x_1, ..., x_n]$. Let $L = \text{lcm}(\text{lp}(f), \text{lp}(g))$. The polynomial

$$S(f,g) = \frac{L}{lt(f)}f - \frac{L}{lt(g)}g$$

is called the *S-polynomial* of f and g [AL94].

Definition 11. Given f, g, h in $k[x_1, ..., x_n]$, with $g \neq 0$, we say that f reduces to h modulo g in one step, written,

$$f \xrightarrow{g} h,$$

if and only if $\text{lp}(g)$ divides a non-zero term X that appears in f and

$$h = f - \frac{X}{lt(g)}g$$

[AL94].

Definition 12. Let f, h, and $f_1, ..., f_s$ be polynomials in $k[x_1, ..., x_n]$, with $f_i \neq 0, (1 \leq i \leq s)$, and let $F = \{f_1, ..., f_s\}$. We say that f reduces to h modulo F denoted

$$f \xrightarrow{F}{}_{+} h$$

if and only if there exist a sequence of indices $i_1, i_2, ..., i_t \in \{1, ..., s\}$ and a sequence of polynomials $h_1, ..., h_{t-1} \in k[x_1, ..., x_n]$ such that

$$f \xrightarrow{f_{i_1}} h_1 \xrightarrow{f_{i_2}} h_2 \xrightarrow{f_{i_3}} \dots \xrightarrow{f_{i_{t-1}}} h_{t-1} \xrightarrow{f_{i_t}} h.$$

With these definitions, we can now state a more convenient characterization of a Gröbner basis.

Algorithm 1 Buchberger's Algorithm

INPUT: $F = \{f_1, ..., f_s\} \subseteq k[x_1, ..., x_n]$with $f_i \neq 0 (1 \leq i \leq s)$
OUTPUT: $G = \{g_1, ..., g_t\}$, a Gröbner basis for $\langle f_1, ..., f_s \rangle$
INITIALIZATION: $G := F$, $\mathcal{G} := \{\{f_i, f_j\} | f_i \neq f_j \in G\}$
WHILE $\mathcal{G} \neq \emptyset$ **DO**
Choose any $\{f, g\} \in \mathcal{G}$
$\mathcal{G} := \mathcal{G} - \{\{f, g\}\}$
$S(f, g) \xrightarrow{G}_+ h$ where h is reduced with respect to G
IF $h \neq 0$ **THEN**
$\mathcal{G} := \mathcal{G} \cup \{\{u, h\} | \text{for all } u \in G\}$
$G := G \cup \{h\}$

Theorem 13. *(Buchberger) Let $G = \{g_1, ..., g_t\}$be a set of non-zero polynomials in $k[x_1, ..., x_n]$. Then G is a Gröbner basis for the ideal $I = \langle g_1, ..., g_t \rangle$ if and only if for all $i \neq j$,*

$$S(g_i, ..., g_j) \xrightarrow{G}_+ 0$$

[AL94].

4. Buchberger's Algorithm

Computations of the solution of a system can be made easier if the system is transformed in some way to a different system that is easier to solve, while maintaining the same solutions. Recall Gauss-Jordan elimination, where by altering the equations in the system, we are able to transform the system of a linear equations into row-echelon form. The system yields the same solution but is much easier to solve with back-substitution.

We want to use Buchberger's Algorithm in order to compute a Gröbner basis for a set of polynomials that is easier to solve. Algorithm 1 describes Buchberger's Algorithm.

We will do an example of finding a Gröbner basis by hand in the two variable case, below.

Example. We will find the Gröbner basis for the system of polynomials $\{yx - x, y^2 - x\}$ by hand using the algorithm.

We begin with the set $F = \{yx - x, y^2 - x\}$. Our algorithm calls for us to set our Gröbner basis $G = \{yx - x, y^2 - x\}$ and $\mathcal{G} = \{\{yx - x, y^2 - x\}\}$.

When $\mathcal{G} \neq \emptyset$, we choose $\{yx - x, y^2 - x\} \in \mathcal{G}$. Now, $\mathcal{G} = \mathcal{G} - \{\{yx - x, y^2 - x\}\} = \emptyset$. Next, we take the S-polynomial of $(yx - x, y^2 - x)$, as follows:

$$\begin{aligned} S(yx-x,y^2-x) &= \frac{y^2x}{yx}(yx-x) - \frac{y^2x}{y^2}(y^2-x) \\ &= y(yx-x) - x(y^2-x) \\ &= y^2x - yx - y^2x + x^2 \\ &= -yx + x^2. \end{aligned}$$

We obtain $S(yx-x, y^2-x) = -yx + x^2$.

For the last step of the **WHILE** loop, we must reduce our S-polynomial by G until we obtain some remainder h that cannot be further reduced by G. So, we take

$$\frac{-yx+x^2}{yx-x} = -1 + \frac{x^2-x}{yx-x} \Longrightarrow h = x^2 - x.$$

Since our $h \neq 0$, we edit $\mathcal{G}$ such that $\mathcal{G} = \{\{yx-x, x^2-x\}, \{y^2-x, x^2-x\}\}$. Additionally, we add our new h to our Gröbner basis. So, our new $G = \{yx - x, y^2-x, x^2-x\}$. Since $\mathcal{G} \neq \emptyset$, we begin the **WHILE** loop again.

This time, we choose $\{yx-x, x^2-x\} \in \mathcal{G}$. Now, $\mathcal{G} = \mathcal{G} - \{\{yx-x, x^2-x\}\} = \{\{y^2-x, x^2-x\}\}$. Next, we take the S-polynomial of $(yx-x, x^2-x)$, as follows:

$$\begin{aligned} S(yx-x,x^2-x) &= \frac{yx^2}{yx}(yx-x) - \frac{yx^2}{x^2}(x^2-x) \\ &= x(yx-x) - y(x^2-x) \\ &= yx^2 - x^2 - yx^2 + yx \\ &= yx - x^2 \end{aligned}$$

We obtain $S(yx-x, x^2-x) = yx - x^2$.

For the last step of the **WHILE** loop, we must reduce our S-polynomial by G to again obtain some remainder h that cannot be further reduced by G. So, we take

$$\frac{yx-x^2}{yx-x} = 1 + \frac{-x^2+x}{yx-x} \Longrightarrow h = -x^2 + x.$$

However, this h can be further reduced by a member of G. So, we take again

$$\frac{-x^2+x}{x^2-x} = -1 + 0 \Longrightarrow h = 0.$$

Since we have obtained an $h = 0$, the **WHILE** loop is complete. Thus, the Gröbner basis for $\langle yx-x, y^2-x \rangle$ is $G = \{yx-x, y^2-x, x^2-x\}$. I verified this calculation using Mathematica.

```
cells = Flatten[Table[(w_{i,j} - 1) (w_{i,j} - 2) (w_{i,j} - 3) (w_{i,j} - 4) == 0,
    {i, 1, 4, 1}, {j, 1, 4, 1}]];

rows =
  Flatten[
   Table[{w_{i,1} + w_{i,2} + w_{i,3} + w_{i,4} - 10 == 0, w_{i,1} w_{i,2} w_{i,3} w_{i,4} - 24 == 0},
    {i, 1, 4, 1}]];

cols =
  Flatten[
   Table[{w_{1,j} + w_{2,j} + w_{3,j} + w_{4,j} - 10 == 0, w_{1,j} w_{2,j} w_{3,j} w_{4,j} - 24 == 0},
    {j, 1, 4, 1}]];

subsquares =
  Flatten[Table[{w_{i,j} + w_{i,j+1} + w_{i+1,j} + w_{i+1,j+1} - 10 == 0,
     w_{i,j} w_{i,j+1} w_{i+1,j} w_{i+1,j+1} - 24 == 0}, {i, 1, 3, 2}, {j, 1, 3, 2}]];

eqns = Flatten[Union[cells, rows, cols, subsquares]];

varorder = Flatten[Table[w_{i,j}, {i, 4, 1, -1}, {j, 4, 1, -1}]];


soln = GroebnerBasis[Union[eqns], varorder]
```

Figure 5.1. Sum-Product Shidoku Code

5. Finding a Gröbner Basis Using Mathematica

It becomes evident in the example above how increasingly complex this algorithm becomes when there are more than 2 variables. Next, we will discuss how we found a Gröbner basis for our systems of equations in 16 variables.

I first coded the sum-product Shidoku system. To do so, I input all 40 of the equations mentioned in section 2.1 of this paper into Mathematica, as can be seen in Figure 5.1 (Sum-Product Shidoku Code).

I then used reverse lexicographical term ordering. Because our system of polynomial equations has a finite number of solutions, the reduced Gröbner basis for the ideal generated by these polynomials using the lexicographical term ordering is triangular [ALT10]. Since our basis will be triangular, the first 16 equations will allow us to use back-substitution to solve the system of equations. I then used Mathematica to find the Gröbner basis for the ideal generated by the sum-product Shidoku system. Since no clues were given, this basis represents only the inherent structure of the Shidoku board. The output is a system of 17 polynomials, 16 of which are triangular, will be discussed in the next section.

6. Counting

6.1. Shidoku Counting. By simple counting, we know that the maximum possible number of boards we could have is $4^{16} = 4,294,967,296$[ALT10]. However, because of our constraints, we can infer that there are not nearly that many possibilities.

The figure below represents the number of choices we would have if we began to count the possible number of boards by hand.

(4)	(3)	(2)	(1)
(2)	(1)		

Beginning in the upper left-hand corner, we have four possibilities for $w_{1,1}$, three choices for $w_{1,2}$, two choices for $w_{1,3}$, and so on. Thus, there are $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 = 48$ possible combinations for the first 6 squares. Let us look below at the Gröbner basis for the Sum-Product Shidoku system:

(1) $w_{1,1}^4 - 10w_{1,1}^3 + 35w_{1,1}^2 - 50w_{1,1} + 24$

(2) $w_{1,2}^3 + w_{1,1}w_{1,2}^2 - 10w_{1,2}^2 + w_{1,1}^2 w_{1,2} - 10w_{1,1}w_{1,2} +$ lower terms

(3) $w_{1,3}^2 + w_{1,2}w_{1,3} + w_{1,1}w_{1,3} - 10w_{1,3} + w_{1,2}^2 +$ lower terms

(4) $w_{1,4} + w_{1,3} + w_{1,2} + w_{1,1} - 10$

(5) $w_{2,1}^2 + w_{1,2}w_{2,1} + w_{1,1}w_{2,1} - 10w_{2,1} +$ lower terms

(6) $w_{2,2} + w_{2,1} + w_{1,2} + w_{1,1} - 10$

(7) $w_{2,3}^2 - w_{1,2}w_{2,3} - w_{1,1}w_{2,3} + w_{1,1}w_{1,2}$

(8) $w_{2,4} + w_{2,3} - w_{1,2} - w_{1,1}$

(9) $w_{3,1}^2 + w_{2,1}w_{3,1} + w_{1,1}w_{3,1} - 10w_{3,1} - w_{1,2}w_{2,1} +$ lower terms

(10) $9w_{2,3}w_{3,2} - 10w_{1,1}^3 w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} + 75w_{1,1}^2 w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} +$lower terms

(11) $w_{3,2}^2 - w_{2,1}w_{3,2} - w_{1,1}w_{3,2} + w_{1,1}w_{2,1}$

(12) $18w_{3,3} - 10w_{1,1}^2 w_{1,2} w_{1,3} w_{2,1} w_{3,2} + 50w_{1,1} w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} +$ lower terms

(13) $18w_{3,4} + 10w_{1,1}^2 w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} - 50w_{1,1} w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} +$ lower terms

(14) $w_{4,1} + w_{3,1} + w_{2,1} + w_{1,1} - 10$

(15) $w_{4,2} + w_{3,2} - w_{2,1} - w_{1,1}$

(16) $18w_{4,3} + 10w_{1,1}^2 w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} - 50w_{1,1} w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} +$ lower terms

(17) $18w_{4,4} - 10w_{1,1}^2 w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} + 50w_{1,1} w_{1,2}^2 w_{1,3} w_{2,1} w_{3,2} +$ lower terms

We can see that 4, 3, 2, 1, 2, and 1 are the powers of the leading power products on equations (1), (2), (3), (4), (5), and (6) of our Gröbner basis. This is not a coincidence, and we will discuss how these results can be used in the end of this section. When finding the solutions to equation (1) when it is equal to zero, we know that we have four possible solutions if there are no repeated roots. Since our basis is triangular, this pattern continues so that equation (2) has 3 possible solutions, equation (3) has 2 possible solutions, and so on. So, to find out how many possible solutions there are for the first 6 equations in our Gröbner basis, we can simply multiply to get $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 = 48$ possible solutions for the first 6 variables $\{w_{1,1}, w_{1,2}, w_{1,3}, w_{1,4}, w_{2,1}, w_{2,2}\}$.

This logic seems to suggest that we could multiply all of the leading term degrees of the Gröbner basis to obtain $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 384$ possible Shidoku boards. However, let us take a closer look at our Gröbner basis. In particular, let us inspect equation number (10). Its leading term contains both $w_{2,3}$and $w_{3,2}$. This particular equation represents a branching effect that will occur when counting the number of possible Shidoku boards. I will introduce this "branching effect" first with a concrete example, then introduce how it will affect our count involving the Gröbner basis.

Since the naming of values is arbitrary, let us begin with one of the 48 Shidoku puzzles, with the values as shown below.

1	2	3	4
3	4		

From here, we will further affect the number of possible boards in our choosing of the next several points. We can see, that if we were to begin to solve this Shidoku puzzle, the value of $w_{2,3}$ can either equal 1 or 2.

If $w_{2,3} = 1$, we can determine that $w_{2,4} = 2$.

1	2	3	4
3	4	1	2

From there, we are presented with another choice at $w_{3,1}$ where it can either equal 2 or 4.

1	2	3	4
3	4	1	2
2		4	
4		2	

1	2	3	4
3	4	1	2
4		2	
2		4	

From here, there is only one more choice left to be made, in cell $w_{3,2}$. Choosing either 1 or 3 will uniquely determine the rest of the board. Thus, for $w_{2,3} = 1$, we have four ways that the board could be solved. Therefore, there are $48 \cdot 1 \cdot 2 \cdot 2 = 192$ different Shidoku boards that are equivalent to those found in the calculation where $w_{2,3} = 1$.

What about the case where $w_{2,3} = 2$? We have the following board.

1	2	3	4
3	4	2	1

Again, we have a choice at $w_{3,1}$, where it can be equal to 2 or 4.

1	2	3	4
3	4	2	1
2	1	4	3
4	3	1	2

1	2	3	4
3	4	2	1
4	3	1	2
2	1	4	3

From here, it can easily be seen that both boards are completely determined. Thus, for $w_{2,3} = 1$, we have two ways that the board could be solved. Therefore, there $48 \cdot 1 \cdot 2 = 96$ different Shidoku boards that are equivalent to the case where $w_{2,3} = 2$.

This gives us a total of $192 + 96 = 288$ possible Shidoku boards [AL94] .

We are able to count the number of Shidoku boards by hand. However, we can infer that the "branching" is much more complex in the Sudoku case, where we have a 9×9 grid. Thus, we need to develop a way of counting using our Gröbner basis.

To do this, we would like to count the number of power products that are not divisible by any of the leading power products of G. This is due to the fact that our ideal I is zero-dimensional [ALT10], meaning there are only a finite number of solutions to the system of polynomial equations.

Theorem 14. *(Finiteness Theorem) Let $k \subset \mathbb{C}$ be a field, and let $I \subset k[x_1, ..., x_n]$ be an ideal. Then the following conditions are equivalent:*

- *The algebra $A = k[x_1, ..., x_n]/I$ is a finite-dimensional over k.*
- *The variety $\mathbf{V}(I) \subset \mathbb{C}^n$ is a finite set.*
- *If G is a Gröbner basis for I, then for each i, $1 \leq i \leq n$, there is an $m_i \geq 0$ such that $x_i^{m_i}$ =Lt(g) for some $g \in G$.*[AL94]

An ideal satisfying any of the above conditions is said to be zero-dimensional. Thus, A is a finite-dimensional algebra if and only if I is a zero-dimensional ideal.

Let $k[x_1, ..., x_n]/I$ be a vector space representing the set of all cosets of I . Since our ideal is zero-dimensional, we can refer to Proposition 2.1.6 in [AL94] that states the following:

Proposition 15. *Let $G = \{g_1, ..., g_m\}$ be a Gröbner basis for an ideal I. A basis for the k vector space $k[x_1, ..., x_n]/I$ consists of the cosets of all the power products X such that lp(g_i) does not divide X for all $i = 1, 2, ...m$* [AL94].

For us, this means that the basis for the vector space $\mathbb{Q}[w_{1,1}, ..., w_{4,4}]/I$ is all of the cosets such that the leading power product of any polynomial in G is not divisible by any other leading power product of G [ALT10].

An ideal J is said to be a radical ideal if for some power of a polynomial $f^n \in J$, then $f \in J$. For example, the ideal $I = \langle x^2, y^2 \rangle$ in $\mathbb{R}[x, y]$ is not radical, since neither x nor y is in the ideal I. However, the ideal $J = \langle x, y \rangle$ is a radical ideal, since x and y are in the ideal J. We also know that our ideal is radical [ALT10], which allows us to assume that we have no repeated roots in our Gröbner basis. Let us use the following Theorem to show by example why we can assume that we have no repeated roots in our Gröbner basis.

Theorem 16. *A set of non-zero polynomials $G = \{g_1, ..., g_t\}$ contained in an ideal I, is a Gröbner basis for I if and only if for all $f \in I$ such that $f \neq 0$, there exists $i \in \{1, ..., t\}$ such that $lp(g_i)$ divides $lp(f)$*[AL94].

Suppose that $g = xy^2$ is a polynomial with a repeated root in a Gröbner basis G. We know that $f = x \cdot g = x^2y^2 \in I$. Since I is radical, and $x^2y^2 \in I$, then $xy \in I$. However, per the Theorem above, g must divide f. Thus, our Gröbner basis must have no repeated roots.

If we know that we have a zero-dimensional, radical ideal, we can use a Theorem from [CLO98].

Theorem 17. *Let I be a zero-dimensional ideal in $\mathbb{C}[x_1, ..., x_n]$, and let $A = \mathbb{C}[x_1, ..., x_n]/I$. Then $dim_{\mathbb{C}}(A)$ is greater than or equal to the number of points in the solution to the system of polynomials. Moreover, equality occurs if and only if I is a radical ideal* [CLO98].

Thus, we know that the dimension of $\mathbb{Q}[w_{1,1}, ..., w_{4,4}]/I$ is the number of solutions to the system of polynomials [CLO98].

Now, we need to use the Gröbner basis in order to count the number of power products that are not divisible by any leading power products of G. We must remove any power products that are divisible by any leading power products of G. Allowable power products will be items not divisible by any leading power product of G, for example $w_{1,1}^0, w_{1,1}, w_{1,1}^2, w_{1,1}^3, w_{1,1}w_{1,2}, w_{1,1}^2w_{1,2}$ and so on. How do we know the number of power products we have? We can count the number of these power products using the form $w_{1,1}^{r_{1,1}}w_{1,2}^{r_{1,2}} \cdots w_{4,4}^{r_{4,4}}$, where $r_{1,1}, ..., r_{4,4}$ represent the power of each variable. For instance, if we do not want the power product to be divisible by $w_{1,1}^4$(the leading power product of (1)), we have 4 choices of $r_{1,1}$ which would not make the power product divisible by $w_{1,1}^4$(namely, $w_{1,1}^0, w_{1,1}^1, w_{1,1}^2$, and $w_{1,1}^3$). Continuing in this manner, we have $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 384$ choices so far [ALT10].

To remove any power products that are divisible by any leading power products of G, we take another look at the polynomial (10) from our Gröbner basis, the one whose leading term contained both $w_{2,3}$ and $w_{3,2}$. We need only concern ourselves with this polynomial since each of the other 16 leading power products are in only one distinct variable, and are therefore, not divisible by any other power products of G. The non-allowable power products (those that will be divisible by some other leading power product of G) are of the form $w_{1,1}^{r_{1,1}} w_{1,2}^{r_{1,2}} \cdots w_{4,4}^{r_{4,4}}$. We know any non-allowable power products will be of the form $w_{1,1}^{r_{1,1}} w_{1,2}^{r_{1,2}} \cdots w_{4,4}^{r_{4,4}}$ where $r_{2,3} = r_{3,2} = 1$ (since $r_{2,3}^1$ and $r_{3,2}^1$ would create a power product that is divisible by another power product). Thus, there are $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 96$ power products which are divisible by a leading power product that we must remove. Thus, we have $384 - 96 = 288$ different possible Shidoku boards, as expected [ALT10].

6.2. **Roots of Unity Shidoku System.** I next decided to move to the Roots of Unity and Boolean Shidoku systems, since the board count should remain the same no matter what representation of the Shidoku board that we use. I began with Roots of Unity and used the 72 (see Section 2.2) polynomial equations in Mathematica as before, as can be seen in Figure 6.1 (Roots of Unity Shidoku Code). I coded the equations, used reverse lexicographical term ordering, and attempted to find the Gröbner basis for the ideal generated by the Roots of Unity system. However, since the Roots of Unity system requires almost double the amount of input equations as the Sum-Product system, Mathematica was not able to find a Gröbner basis without the use of clues. We know from our previous calculations that we have 288 total possible Shidoku boards. If we insert the first row of clues such that $w_{1,1} = 1, w_{1,2} = -1, w_{1,3} = \mathrm{I}$, and $w_{1,4} = -\mathrm{I}$, we can assert that there are $\frac{288}{4!} = 12$ possible boards by simple counting principles. Thus, I found a Gröbner basis using both the equations and clues listed above in Figure 6.1 (Roots of Unity Shidoku Code). The output for this system is as follows:

(1) $w_{1,1} - 1$
(2) $w_{1,2} + 1$
(3) $w_{1,3} - i$
(4) $w_{1,4} + i$
(5) $w_{2,1}^2 + 1$
(6) $w_{2,2} + w_{2,1}$
(7) $w_{2,3}^2 - 1$
(8) $w_{2,4} + w_{2,3}$
(9) $w_{3,1}^2 + w_{2,1}w_{3,1} + w_{3,1} + w_{2,1}$
(10) $w_{2,3}w_{3,2} + iw_{2,1}w_{3,2} + w_{2,3}w_{3,1} + iw_{2,1}w_{3,1}$
(11) $w_{3,2}^2 - w_{2,1}w_{3,2} - w_{3,2} - w_{2,1}$
(12) $2w_{3,3} + iw_{2,1}w_{3,2} + w_{3,2} + w_{2,1}w_{2,3}w_{3,1} + w_{2,3}w_{3,1} + (1-i)w_{3,1}$+lower terms
(13) $2w_{3,4} - iw_{2,1}w_{3,2} + w_{3,2} - w_{2,1}w_{2,3}w_{3,1} - w_{2,3}w_{3,1} + (1+i)w_{3,1}$+lower terms

(14) $w_{4,1} + w_{3,1} + w_{2,1} + 1$

(15) $w_{4,2} + w_{3,2} - w_{2,1} - 1$

(16) $2w_{4,3} - iw_{2,1}w_{3,2} - w_{3,2} - w_{2,1}w_{2,3}w_{3,1} - w_{2,3}w_{3,1} - (1-i)w_{3,1}+$lower terms

(17) $2w_{4,4} + iw_{2,1}w_{3,2} - w_{3,2} + w_{2,1}w_{2,3}w_{3,1} + w_{2,3}w_{3,1} - (1+i)w_{3,1}+$lower terms

The total number of power products for this basis is $1 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 16$. We then remove our non-allowable power products: $1 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 4$. We find by using the leading power products in the same way as before, there are $16 - 4 = 12$ possible Shidoku boards of this form, as expected.

```
cells = Flatten[Table[w_{i,j}^4 == 1, {i, 1, 4, 1}, {j, 1, 4, 1}]];

rows =
  Flatten[
   Table[{(w_{i,1} + w_{i,2}) (w_{i,1}^2 + w_{i,2}^2) == 0,
     (w_{i,1} + w_{i,3}) (w_{i,1}^2 + w_{i,3}^2) == 0, (w_{i,1} + w_{i,4}) (w_{i,1}^2 + w_{i,4}^2) == 0,
     (w_{i,2} + w_{i,3}) (w_{i,2}^2 + w_{i,3}^2) == 0, (w_{i,2} + w_{i,4}) (w_{i,2}^2 + w_{i,4}^2) == 0,
     (w_{i,3} + w_{i,4}) (w_{i,3}^2 + w_{i,4}^2) == 0}, {i, 1, 4, 1}]];

cols =
  Flatten[
   Table[{(w_{1,j} + w_{2,j}) (w_{1,j}^2 + w_{2,j}^2) == 0,
     (w_{1,j} + w_{3,j}) (w_{1,j}^2 + w_{3,j}^2) == 0, (w_{1,j} + w_{4,j}) (w_{1,j}^2 + w_{4,j}^2) == 0,
     (w_{2,j} + w_{3,j}) (w_{2,j}^2 + w_{3,j}^2) == 0, (w_{2,j} + w_{4,j}) (w_{2,j}^2 + w_{4,j}^2) == 0,
     (w_{3,j} + w_{4,j}) (w_{3,j}^2 + w_{4,j}^2) == 0}, {j, 1, 4, 1}]];

subsquares =
  Flatten[{(w_{1,1} + w_{2,2}) (w_{1,1}^2 + w_{2,2}^2) == 0,
    (w_{1,2} + w_{2,1}) (w_{1,2}^2 + w_{2,1}^2) == 0, (w_{1,3} + w_{2,4}) (w_{1,3}^2 + w_{2,4}^2) == 0,
    (w_{1,4} + w_{2,3}) (w_{1,4}^2 + w_{2,3}^2) == 0, (w_{3,1} + w_{4,2}) (w_{3,1}^2 + w_{4,2}^2) == 0,
    (w_{3,2} + w_{4,1}) (w_{3,2}^2 + w_{4,1}^2) == 0, (w_{3,3} + w_{4,4}) (w_{3,3}^2 + w_{4,4}^2) == 0,
    (w_{3,4} + w_{4,3}) (w_{3,4}^2 + w_{4,3}^2) == 0}];

eqns = Flatten[Union[cells, rows, cols, subsquares]];

varaorder = Flatten[Table[w_{i,j}, {i, 4, 1, -1}, {j, 4, 1, -1}]];

   clues = {w_{1,1} = 1, w_{1,2} = -1, w_{1,3} = I, w_{1,4} = -I};

   soln = GroebnerBasis[Union[eqns, clues], varaorder]
```

FIGURE 6.1. Roots of Unity Shidoku Code

6.3. **Boolean Shidoku System.** I ended my explorations of Shidoku puzzles with the Boolean system, and attempted to use the 136 (see Section 2.3) equations to find a Gröbner basis that we could use to count Shidoku boards. This code can be seen in Figure 6.2 (Boolean Shidoku Code). Again, since the number of equations was far too large for Mathematica to properly execute, I added clues to make the top row read across as 1, 2, 3, 4. To do this is the Boolean case, it required us to insert 16 clues such that $w_{1,1,1} = 1, w_{1,1,2} = 0, w_{1,1,3} = 0, w_{1,1,4} = 0, w_{1,2,1} = 0, w_{1,2,2} = 1, w_{1,2,3} = 0, w_{1,2,4} = 0, w_{1,3,1} = 0, w_{1,3,2} = 0, w_{1,3,3} = 1, w_{1,3,4} = 0, w_{1,4,1} = 0, w_{1,4,2} = 0, w_{1,4,3} = 0$, and $w_{1,4,4} = 1$. I found a Gröbner basis using equations and the clues listed above and obtained:

(1) $w_{1,1,1} - 1$
(2) $w_{1,1,2}$
(3) $w_{1,1,3}$
(4) $w_{1,1,4}$
(5) $w_{1,2,1}$
(6) $w_{1,2,2} - 1$
(7) $w_{1,2,3}$
(8) $w_{1,2,4}$
(9) $w_{1,3,1}$
(10) $w_{1,3,2}$
(11) $w_{1,3,3} - 1$
(12) $w_{1,3,4}$
(13) $w_{1,4,1}$
(14) $w_{1,4,2}$
(15) $w_{1,4,3}$
(16) $w_{1,4,4} - 1$
(17) $w_{2,1,1}$
(18) $w_{2,1,2}$
(19) $w_{2,1,3}^2 - w_{2,1,3}$
(20) $w_{2,1,4} + w_{2,1,3} - 1$
(21) $w_{2,2,1}$
(22) $w_{2,2,2}$
(23) $w_{2,2,3} + w_{2,1,3-1}$
(24) $w_{2,2,4} - w_{2,1,3}$
(25) $w_{2,3,1}^2 - w_{2,3,1}$
(26) $w_{2,3,2} + w_{2,3,1} - 1$
(27) $w_{2,3,3}$
(28) $w_{2,3,4}$
(29) $w_{2,4,1} + w_{2,3,1} - 1$

(30) $w_{2,4,2} - w_{2,3,1}$

(31) $w_{2,4,3}$

(32) $w_{2,4,4}$

(33) $w_{3,1,1}$

(34) $w_{3,1,2}^2 - w_{3,1,2}$

(35) $w_{3,1,3} - w_{2,1,3}w_{3,1,2} + w_{3,1,2} + w_{2,1,3} - 1$

(36) $w_{3,1,4} + w_{2,1,3}w_{3,1,2} - w_{2,1,3}$

(37) $w_{2,3,1}w_{3,2,1} - w_{2,1,3}w_{3,2,1} - w_{2,3,1}w_{3,1,2} + w_{2,1,3}w_{3,1,2}$

(38) $w_{3,2,1}^2 - w_{3,2,1}$

(39) $w_{3,2,2}$

(40) $w_{3,2,3} + w_{2,1,3}w_{3,2,1} - w_{2,1,3}$

(41) $w_{3,2,4} - w_{2,1,3}w_{3,2,1} + w_{3,2,1} + w_{2,1,3} - 1$

(42) $w_{3,3,1} - w_{2,1,3}w_{3,2,1} + w_{3,2,1} - w_{2,3,1}w_{3,1,2} + w_{2,1,3}w_{3,1,2} + w_{2,3,1} - 1$

(43) $w_{3,3,2} + w_{2,3,1}w_{3,1,2} - w_{2,3,1}$

(44) $w_{3,3,3}$

(45) $w_{3,3,4} + w_{2,1,3}w_{3,2,1} - w_{3,2,1} - w_{2,1,3}w_{3,1,2}$

(46) $w_{3,4,1} + w_{2,1,3}w_{3,2,1} + w_{2,3,1}w_{3,1,2} - w_{2,1,3}w_{3,1,2} - w_{2,3,1}$

(47) $w_{3,4,2} - w_{2,3,1}w_{3,1,2} + w_{3,1,2} + w_{2,3,1} - 1$

(48) $w_{3,4,3} - w_{2,1,3}w_{3,2,1} + w_{2,1,3}w_{3,1,2} - w_{3,1,2}$

(49) $w_{3,4,4}$

(50) $w_{4,1,1}$

(51) $w_{4,1,2} + w_{3,1,2} - 1$

(52) $w_{4,1,3} + w_{2,1,3}w_{3,1,2} - w_{3,1,2}$

(53) $w_{4,1,4} - w_{2,1,3}w_{3,1,2}$

(54) $w_{4,2,1} + w_{3,2,1} - 1$

(55) $w_{4,2,2}$

(56) $w_{4,2,3} - w_{2,1,3}w_{3,2,1}$

(57) $w_{4,2,4} + w_{2,1,3}w_{3,2,1} - w_{3,2,1}$

(58) $w_{4,3,1} + w_{2,1,3}w_{3,2,1} - w_{3,2,1} + w_{2,3,1}w_{3,1,2} - w_{2,1,3}w_{3,1,2}$

(59) $w_{4,3,2} - w_{2,3,1}w_{3,1,2}$

(60) $w_{4,3,3}$

(61) $w_{4,3,4} - w_{2,1,3}w_{3,2,1} + w_{3,2,1} + w_{2,1,3}w_{3,1,2} - 1$

(62) $w_{4,4,1} - w_{2,1,3}w_{3,2,1} - w_{2,3,1}w_{3,1,2} + w_{2,1,3}w_{3,1,2}$

(63) $w_{4,4,2} + w_{2,3,1}w_{3,1,2} - w_{3,1,2}$

(64) $w_{4,4,3} + w_{2,1,3}w_{3,2,1} - w_{2,1,3}w_{3,1,2} + w_{3,1,2} - 1$

(65) $w_{4,4,4}$

The total number of power products for this basis is $1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot2\cdot1\cdot1\cdot1\cdot1\cdot1\cdot2\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot1\cdot2\cdot1\cdot1\cdot2\cdot1 = 16$. We then remove our non-allowable power

```
cells1 = Flatten[Table[w_{i,j,k} (w_{i,j,k} - 1) == 0, {i, 1, 4, 1},
    {j, 1, 4, 1}, {k, 1, 4, 1}]];

cells2 =
  Flatten[{w_{1,1,1} + w_{1,1,2} + w_{1,1,3} + w_{1,1,4} == 1, w_{1,2,1} + w_{1,2,2} + w_{1,2,3} + w_{1,2,4} == 1,
    w_{1,3,1} + w_{1,3,2} + w_{1,3,3} + w_{1,3,4} == 1, w_{1,4,1} + w_{1,4,2} + w_{1,4,3} + w_{1,4,4} == 1,
    w_{2,1,1} + w_{2,1,2} + w_{2,1,3} + w_{2,1,4} == 1, w_{2,2,1} + w_{2,2,2} + w_{2,2,3} + w_{2,2,4} == 1,
    w_{2,3,1} + w_{2,3,2} + w_{2,3,3} + w_{2,3,4} == 1, w_{2,4,1} + w_{2,4,2} + w_{2,4,3} + w_{2,4,4} == 1,
    w_{3,1,1} + w_{3,1,2} + w_{3,1,3} + w_{3,1,4} == 1, w_{3,2,1} + w_{3,2,2} + w_{3,2,3} + w_{3,2,4} == 1,
    w_{3,3,1} + w_{3,3,2} + w_{3,3,3} + w_{3,3,4} == 1, w_{3,4,1} + w_{3,4,2} + w_{3,4,3} + w_{3,4,4} == 1,
    w_{4,1,1} + w_{4,1,2} + w_{4,1,3} + w_{4,1,4} == 1, w_{4,2,1} + w_{4,2,2} + w_{4,2,3} + w_{4,2,4} == 1,
    w_{4,3,1} + w_{4,3,2} + w_{4,3,3} + w_{4,3,4} == 1, w_{4,4,1} + w_{4,4,2} + w_{4,4,3} + w_{4,4,4} == 1}];

rows =
  Flatten[Table[{w_{i,1,1} w_{i,2,1} + w_{i,1,2} w_{i,2,2} + w_{i,1,3} w_{i,2,3} + w_{i,1,4} w_{i,2,4} == 0,
     w_{i,1,1} w_{i,3,1} + w_{i,1,2} w_{i,3,2} + w_{i,1,3} w_{i,3,3} + w_{i,1,4} w_{i,3,4} == 0,
     w_{i,1,1} w_{i,4,1} + w_{i,1,2} w_{i,4,2} + w_{i,1,3} w_{i,4,3} + w_{i,1,4} w_{i,4,4} == 0,
     w_{i,2,1} w_{i,3,1} + w_{i,2,2} w_{i,3,2} + w_{i,2,3} w_{i,3,3} + w_{i,2,4} w_{i,3,4} == 0,
     w_{i,2,1} w_{i,4,1} + w_{i,2,2} w_{i,4,2} + w_{i,2,3} w_{i,4,3} + w_{i,2,4} w_{i,4,4} == 0,
     w_{i,3,1} w_{i,4,1} + w_{i,3,2} w_{i,4,2} + w_{i,3,3} w_{i,4,3} + w_{i,3,4} w_{i,4,4} == 0}, {i, 1, 4, 1}]];

cols =
  Flatten[Table[{w_{1,j,1} w_{2,j,1} + w_{1,j,2} w_{2,j,2} + w_{1,j,3} w_{2,j,3} + w_{1,j,4} w_{2,j,4} == 0,
     w_{1,j,1} w_{3,j,1} + w_{1,j,2} w_{3,j,2} + w_{1,j,3} w_{3,j,3} + w_{1,j,4} w_{3,j,4} == 0,
     w_{1,j,1} w_{4,j,1} + w_{1,j,2} w_{4,j,2} + w_{1,j,3} w_{4,j,3} + w_{1,j,4} w_{4,j,4} == 0,
     w_{2,j,1} w_{3,j,1} + w_{2,j,2} w_{3,j,2} + w_{2,j,3} w_{3,j,3} + w_{2,j,4} w_{3,j,4} == 0,
     w_{2,j,1} w_{4,j,1} + w_{2,j,2} w_{4,j,2} + w_{2,j,3} w_{4,j,3} + w_{2,j,4} w_{4,j,4} == 0,
     w_{3,j,1} w_{4,j,1} + w_{3,j,2} w_{4,j,2} + w_{3,j,3} w_{4,j,3} + w_{3,j,4} w_{4,j,4} == 0}, {j, 1, 4, 1}]];

subsquares =
  Flatten[{w_{1,1,1} w_{2,2,1} + w_{1,1,2} w_{2,2,2} + w_{1,1,3} w_{2,2,3} + w_{1,1,4} w_{2,2,4} == 0,
    w_{1,2,1} w_{2,1,1} + w_{1,2,2} w_{2,1,2} + w_{1,2,3} w_{2,1,3} + w_{1,2,4} w_{2,1,4} == 0,
    w_{1,3,1} w_{2,4,1} + w_{1,3,2} w_{2,4,2} + w_{1,3,3} w_{2,4,3} + w_{1,3,4} w_{2,4,4} == 0,
    w_{1,4,1} w_{2,3,1} + w_{1,4,2} w_{2,3,2} + w_{1,4,3} w_{2,3,3} + w_{1,4,4} w_{2,3,4} == 0,
    w_{3,1,1} w_{4,2,1} + w_{3,1,2} w_{4,2,2} + w_{3,1,3} w_{4,2,3} + w_{3,1,4} w_{4,2,4} == 0,
    w_{3,2,1} w_{4,1,1} + w_{3,2,2} w_{4,1,2} + w_{3,2,3} w_{4,1,3} + w_{3,2,4} w_{4,1,4} == 0,
    w_{3,3,1} w_{4,4,1} + w_{3,3,2} w_{4,4,2} + w_{3,3,3} w_{4,4,3} + w_{3,3,4} w_{4,4,4} == 0,
    w_{3,4,1} w_{4,3,1} + w_{3,4,2} w_{4,3,2} + w_{3,4,3} w_{4,3,3} + w_{3,4,4} w_{4,3,4} == 0}];

eqns = Flatten[Union[cells1, cells2, rows, cols, subsquares]];

varaorder = Flatten[Table[w_{i,j,k}, {i, 4, 1, -1}, {j, 4, 1, -1}, {k, 4, 1, -1}]];

clues = {w_{1,1,1} = 1, w_{1,1,2} = 0, w_{1,1,3} = 0, w_{1,1,4} = 0, w_{1,2,1} = 0,
   w_{1,2,2} = 1, w_{1,2,3} = 0, w_{1,2,4} = 0, w_{1,3,1} = 0, w_{1,3,2} = 0 , w_{1,3,3} = 1,
   w_{1,3,4} = 0, w_{1,4,1} = 0, w_{1,4,2} = 0, w_{1,4,3} = 0, w_{1,4,4} = 1};

soln = GroebnerBasis[Union[eqns, clues], varaorder]
```

products: $1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 \cdot 2 \cdot 1 = 4$. We find by using the leading power products in the same way as before, there are $16 - 4 = 12$ possible Shidoku boards of this form, as expected.

6.4. **Applications for Sudoku.** The last natural extension of these findings is to see how we can apply our knowledge to the Sudoku case. We began with the Sum-Product Sudoku System. The Sum-Product Sudoku code can be seen in Figure 6.4 (Sum-Product Shidoku Code). This system contains 135 equations. It was obtained using the same method as the Sum-Product Shidoku code, but with slightly different numbers to accommodate to the 9×9 grid. We were required to use the number set $\{-2, -1, 1, 2, 3, 4, 5, 6, 7\}$ instead of the number set $\{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ as would be expected. This is because there is more than one way to combine the numbers $1 - 9$ and obtain a sum of 45 and a product of 362,880, namely the set $\{1, 2, 4, 4, 4, 5, 7, 9, 9\}$. By using the number set $\{-2, -1, 1, 2, 3, 4, 5, 6, 7\}$, we have a system of numbers whose sum is 25 and whose product is 10,080. This sum and product can only be obtained from this set of numbers by using each number in the set exactly once according to [ALT10].

I coded the Boolean Sudoku in much the same way that I did the Boolean Shidoku, with slight accommodations for the size. This resulted in 1,620 equations.

In the Sudoku case for both the Sum-Product and Boolean systems, not only are there several more equations, there are several more variables and much longer equations contained in these polynomial representations. We attempted to generate Gröbner bases for both of these systems, but the computing power needed was too large. We also attempted to put in clues to a Sudoku puzzle and attempted to use it as a puzzle solver, since with these 37 clues, the algorithm would have had to compute significantly less. We tried doing this for both the Sum-Product and Boolean Sudoku systems that we had created. However, in Mathematica, we could not get either to run.

A point of interest in finding Gröbner bases for Sudoku puzzles may lie in the inherent structure of Buchberger's Algorithm. In the algorithm, there are two places where choices are made: the order in which the polynomials are input, which affects the application of the Division Algorithm, and in the **WHILE** loop where we compute S-polynomials, we choose $\{f, g\}$ at random. Either of these choices could lead us to a different Gröbner basis [AL94].

Currently, it has been shown that there are 6,670,903,752,021,072,936,960 possible Sudoku boards [ALT10]. This method of counting using Gröbner bases described in this paper may be helpful in developing a structured way to count Sudoku

```
cells =
  Flatten[
   Table[(w_{i,j} + 2) (w_{i,j} + 1) (w_{i,j} - 1) (w_{i,j} - 2) (w_{i,j} - 3)
      (w_{i,j} - 4) (w_{i,j} - 5) (w_{i,j} - 6) (w_{i,j} - 7) == 0, {i, 1, 9, 1},
    {j, 1, 9, 1}]];

rows =
  Flatten[
   Table[
    {w_{i,1} + w_{i,2} + w_{i,3} + w_{i,4} + w_{i,5} + w_{i,6} + w_{i,7} + w_{i,8} + w_{i,9} - 25 == 0,
     w_{i,1} w_{i,2} w_{i,3} w_{i,4} w_{i,5} w_{i,6} w_{i,7} w_{i,8} w_{i,9} - 10080 == 0} ,
    {i, 1, 9, 1}]];

cols =
  Flatten[
   Table[
    {w_{1,j} + w_{2,j} + w_{3,j} + w_{4,j} + w_{5,j} + w_{6,j} + w_{7,j} + w_{8,j} + w_{9,j} - 25 ==
      0, w_{1,j} w_{2,j} w_{3,j} w_{4,j} w_{5,j} w_{6,j} w_{7,j} w_{8,j} w_{9,j} - 10080 == 0},
    {j, 1, 9, 1}]];

subsquares =
  Flatten[
   Table[
    {w_{i,j} + w_{i,j+1} + w_{i,j+2} + w_{i+1,j} + w_{i+1,j+1} + w_{i+1,j+2} + w_{i+2,j} +
       w_{i+2,j+1} + w_{i+2,j+2} - 25 == 0,
     w_{i,j} w_{i,j+1} w_{i,j+2} w_{i+1,j} w_{i+1,j+1} w_{i+1,j+2} w_{i+2,j} w_{i+2,j+1} w_{i+2,j+2} -
       10080 == 0}, {i, 1, 7, 3}, {j, 1, 7, 3}]];

eqns = Flatten[Union[cells, rows, cols, subsquares]];

varorder =
  Flatten[Table[w_{i,j}, {i, 9, 1, -1}, {j, 9, 1, -1}]];
```

FIGURE 6.3. Sum-Product Sudoku Code

boards. In addition to counting boards, it should be noted that a Gröbner basis can be used as a Shidoku/Sudoku "solver" given the correct number of clues. It has been conjectured that the minimum number of given values that can completely determine a Sudoku board is 17. This is another opportunity where, with further research, a Gröbner basis could assist us in answering questions related to Sudoku puzzles [ALT10]

References

[ALT10] Arnold, Elizabeth, Stephen Lucas, and Laura Taalman, Gröbner Basis Representations of Sudoku, *The College Mathematics Journal* 41.2 (2010): 101-12.

[Fra03] Fraleigh, John B., *A First Course in Abstract Algebra*, 7th ed., Addison-Wesley, 2003.

[Rot06] Rotman, Joseph J., *A First Course in Abstract Algebra: with Applications*, 3rd ed., Upper Saddle River, NJ: Pearson Prentice Hall, 2006.

[AL94] Adams, William and Philippe Loustaunau, *An Introduction to Gröbner Bases*, American Mathematical Society, 1994.

[CLO98] D. Cox, J. Little, and D. O'Shea, *Using Algebraic Geometry*, Springer-Verlag, New York, 1998.