SJÄLVSTÄNDIGA ARBETEN I MATEMATIK

MATEMATISKA INSTITUTIONEN, STOCKHOLMS UNIVERSITET

En algebraisk framställning av sudoku i polynomringen $\mathbb{Z}_2[x]$

av

Matilda Steffner

2025 - No K3

En algebraisk framställning av sudoku i polynomringen $\mathbb{Z}_2[x]$

Matilda Steffner

Självständigt arbete i matematik 15 högskolepoäng, grundnivå

Handledare: Samuel Lundqvist

2025

Abstract

I ett klassiskt sudoku existerar exakt en unik lösning. Om vi bortser från denna restriktion och tillåter att flera lösningar är möjliga, hur kan vi bestämma hur många lösningar som existerar? I detta arbete söks frågan besvaras genom att modellera sudokus mindre komplexa syskon shidoku i polynomringen $\mathbb{Z}_2[\boldsymbol{x}]$. Därefter används Gröbnerbaser för att besvara systerfrågan: Hur kan vi bestämma antalet möjliga lösningar till ett shidoku? Förhoppningen i detta arbete är att resultatet för shidoku ska kunna extrapoleras till sudoku.

Abstract

In classical sudoku there exists exactly one unique solution. If we allow more than one solution to be possible the following question is raised: How can we calculate the number of possible solutions? In this paper we try answering the question by first modeling sudoku's lesser sibling shidoku in the polynomial ring $\mathbb{Z}_2[\boldsymbol{x}]$. After this we make use of Gröbner basis to answer the similar question: How can we calculate the number of possible solutions to a shidoku? The expectation is to be able to extrapolate the result for shidoku to sudoku.

Innehåll

1 Bakgrund

”Gör så här: Fyll i de tomma rutorna. Alla siffror, från 1 till 9 måste finnas med i varje vågrät rad, varje lodrät rad och vara markerade i varje låda (3×3 rutor).” Så beskrivs reglerna till det klassiska spelet sudoku i tidningen Korsord som utgavs av Expressen, GT och Kvällsposten vecka 30 2024 [5]. Dessa väldigt enkla regler till trots kan sudoku göras väldigt avancerat och ett ändlöst antal variationer av reglerna existerar.

Med dessa regler på plats kan vi ställa oss följande fråga: Hur många olika lösningar finns till ett sudoku? Frågan har olika svar beroende på hur man betraktar begreppet ”olika”. De två ifyllda sudokunäten i figur 1 verkar vara två olika lösningar.

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

(a) En lösning

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

(b) En annan lösning?

Figur 1: Två ”olika” sudokulösningar

Vid närmare inspektion dock ser vi att brädena är identiska förutom det faktum att ettorna har bytt plats med tvåorna. De två brädena ingår i samma symmetrigrupp eftersom de har samma struktur. Så ska de två lösningarna i figur 1 räknas som olika lösningar eller som en och samma lösning?

Vi kan också ifrågasätta vad vi menar med ”sudoku”. Menar vi med detta begrepp den tänkta definitionen, nämligen ett rutnät som är ifyllt på så sätt att det bara existerar en lösning? I sådant fall är frågan inte så värst intressant eftersom vi då redan vet svaret. Menar vi istället ett rutnät som är partiellt ifyllt så att flera lösningar existerar, eller som kanske är helt tomt blir frågan mer spännande.

Frågan kan alltså tolkas på några olika sätt men i det som följer kommer vi att besvara följande fråga: ”Hur många olika lösningar, inklusive lösningar som ingår i samma symmetrigrupp, finns givet ett tomt rutnät, eller ett rutnät som är partiellt ifyllt på så sätt att flera lösningar är möjliga?”. Vi kommer också se att metoden som tillämpas för att beräkna detta kan ge oss svar på om sudokunätet är ett regelrätt sudokupussel, dvs om det bara existerar en lösning.

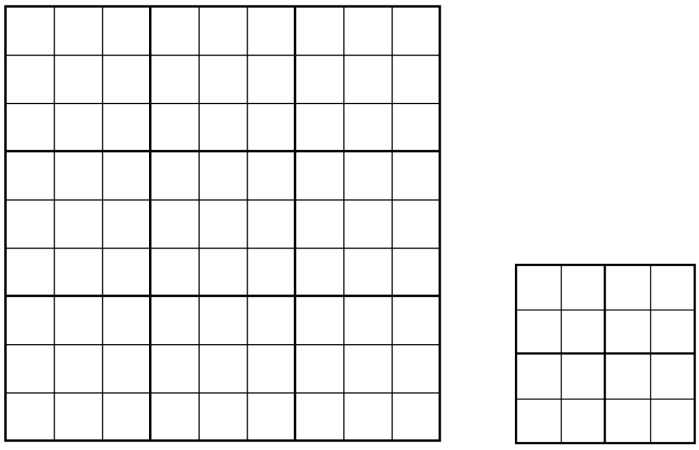
2 Introduktion

Som antytt i Bakgrunden finns ett kriterium som alla sudokunät måste uppfylla för att få kallas ett "sudoku", nämligen att nätet är partiellt ifyllt på så sätt att det existerar en unik lösning. I detta arbete kommer vi dock framför allt att intressera oss för tomma nät, där det uppenbarligen existerar mer än en lösning. Fram till nu har olika begrepp använts lite hipp som happ men för att det inte ska uppstå några missförstånd i det som följer är några definitioner på sin plats.

Med "sudokunät" menas ett tomt eller partiellt ifyllt rutnät så att antingen ingen, en eller flera lösningar existerar, med "sudokupussel" menas ett sudokunät som är partiellt ifyllt så att det existerar en unik lösning och med "lösning" eller "sudokulösning" menas ett sudokunät som är fullständigt ifyllt i enlighet med reglerna för sudoku.

Ett sudokunät består som bekant av ett rutnät med 9×9 celler, och är indelat i 9 boxar med 3×3 celler i vardera enligt figur 2(a). Innan vi ger oss på att modellera reglerna för sudoku börjar vi med att modellera reglerna till sudokus mindre komplexa syskon shidoku. Förhoppningen är att kunna extrapolera modelleringen till det mer komplexa sudokut. Ett shidokunät består av ett rutnät med 4×4 celler och är indelat i 4 boxar med 2×2 celler enligt figur 2(b).

Analogt med definitionerna ovan kommer jag i det som föjer att med "shidokunät" referera till ett tomt eller partiellt ifyllt rutnät så att antingen ingen, en eller flera lösningar existerar, med "shidokupussel" referera till ett shidokunät som är partiellt ifyllt så att det existerar en unik lösning och med "lösning" eller "shidokulösning" referera till ett shidokunät som är fullständigt ifyllt i enlighet med reglerna för shidoku.



(a) Sudokunät (b) Shidokunät

Figur 2: De två brädenas struktur

Reglerna för shidoku är mycket lika de för sudoku. Nu finns det inga shidokun i tidningen Korsord men om det hade funnits några hade reglerna kanske

beskrivits såhär: "Gör så här: Fyll i de tomma rutorna. Alla siffror, från 1 till 4 måste finnas med i varje vågrät rad, varje lodrät rad och vara markerade i varje låda (2×2 rutor)."

3 Algebra

För att beräkna antalet lösningar till ett givet shidokunät kommer vi att använda oss av ideal och Gröbnerbaser men en hel del algebra krävs innan vi kan gå in på dessa koncept. Utläggningen som följer i sektioner 3.1-3.2 följer tätt kapitel 2 i Sass (2011) [3].

3.1 Polynomringar

Definition 3.1 (Polynomring)**.** *Låt K vara en kropp. Polynomringen $K[\boldsymbol{x}]$, där $\boldsymbol{x} = (x_1, x_2, ..., x_n)$ definieras som mängden av polynom i variablerna $x_i \in \boldsymbol{x}$ med koefficienter i K där ringoperationerna är vanlig polynomaddition och polynommultiplikation.*

Några exempel på polynomringar är $\mathbb{R}[x]$ och $\mathbb{Q}[y, z]$. Låt oss ta en titt på dessa polynomringar en i taget.

Exempel 3.1. *$\mathbb{R}[x]$ är ringen med polynom i variabeln x med koefficienter i $\mathbb{R}$, dvs ringen med följande mängd av polynom $\{c_0 + c_1x + c_2x^2 + ... + c_nx^n : c_0, c_1, ..., c_n \in \mathbb{R}, n \in \mathbb{N}\}$. Exempelvis har vi att $p_1 = 1,43+3,1x-0,22x^{49} \in \mathbb{R}[x]$ eftersom p_1 är ett polynom i den enda variabeln x och alla koefficienter ligger i $\mathbb{R}$. Däremot har vi $p_2 = i + 4,444x^{12} - 9x^{17} \notin \mathbb{R}[x]$ eftersom $i \notin \mathbb{R}$ så alla koefficienterna ligger inte i $\mathbb{R}$.*

$\mathbb{Q}[y, z]$ är ringen med polynom i variblerna y, z med koefficienter i $\mathbb{Z}$. Ringen består alltså av alla linjärkombinationer av följande mängd av polynom $\{cy^\alpha z^\beta : c \in \mathbb{Q}, \alpha, \beta \in \mathbb{N}\}$. Exempelvis har vi att $p_1 = -62y+15yz^3 \in \mathbb{Q}[y, z]$ eftersom p_1 är ett polynom i de två variablerna y, z och alla koefficienter ligger i $\mathbb{Q}$. Däremot har vi $p_2 = 3 + 52xy \notin \mathbb{Q}[y, z]$ eftersom p_2 är ett polynom i variablerna x, y och bara variablerna y, z är tillåtna.

Vi kommer att arbeta i en polynomring över kroppen $\mathbb{Z}_2$ vilket ger oss vissa intressanta resultat. För det första behöver vi bara förhålla oss till de två koefficienterna 0 och 1 eftersom dessa är de enda elementen i $\mathbb{Z}_2$. Dessutom har vi att alla exponenter kan reduceras till 1 eftersom de två ekvationerna $x = 0$ och $x^n = 0$ har exakt samma lösningsmängd för alla $n \in \mathbb{N}$. Detsamma gäller för de två ekvationerna $x = 1$ och $x^n = 1$.

Vi kommer dock inte att intressera oss för alla polynom i den aktuella ringen, utan bara ett litet urval av dessa.

Definition 3.2 (Ideal)**.** *Låt R vara en ring. Ett ideal I är en delmängd av R sådan att*

(i) $\forall a, b \in I : a + b \in I$

(ii) $\forall a \in I, \forall c \in R : ac \in I$

Antag att vi har ett ideal I. Definitionen ovan säger oss att alla möjliga kombinationer av elementen i I också är element i I. Så exempelvis om $2 \in I$ och $4 \in I$ så har vi även $2 + 4 = 6 \in I$. Notera dock att det är nödvändigt att specificera vilken ring I är ett ideal över ty annars är det omöjligt att veta vilka element som ligger i I. Antag återigen att $2 \in I$. Är $\frac{2}{3}$ ett element i I? Det är omöjligt att säga om vi inte vet vilken ring I är ett ideal över. Om $R = \mathbb{Q}$ så har vi att $\frac{2}{3} \in I$ ty $2 \in I$ och $\frac{1}{3} \in \mathbb{Q}$. Men om $R = \mathbb{Z}$ så gäller $\frac{2}{3} \notin I$ ty visserligen har vi $2 \in I$ men det existerar inget element x i $\mathbb{Z}$ så att det för någon multipel av 2 gäller: $2n \cdot x = \frac{2}{3}$.

Låt säga att vi vill betrakta något ideal över någon kropp. Eftersom idealet innehåller en så pass stor mängd, och i många fall en oändligt stor mängd, element är det inte gångbart att rada upp alla dessa för att betrakta idealet. Som tur är räcker det att titta på idealets generatorer.

Definition 3.3. *Ett ideal I över R sägs genereras av elementen $p_1, p_2, ..., p_n$ om det för alla element $p_m \in I$ $(m > n)$ gäller att p_m är en kombination av $p_1, p_2, ..., p_n$. Dvs om $p_m = c_1p_1 + c_2p_2 + ... + c_np_n$, med $c_1, c_2, ..., c_n \in R$. Detta skrivs $I = \langle p_1, p_2, ..., p_n \rangle$*

Vi har tittat kort på hur ideal ser ut över ringar som bara innehåller tal men hur förhåller sig ideal till polynomringar?

Exempel 3.2. *Låt $I = \langle x^2, x^2 - 1 \rangle$ vara ett ideal över $\mathbb{Z}[x]$. Då gäller att även alla kombinationer av x^2 och $x^2 - 1$ är element i I, dvs*

$$c_1x^2 + c_2(x^2 - 1) \in I$$

Men eftersom $c_1, c_2 \in \mathbb{Z}[x]$ har vi att c_1 och c_2 kan vara vilka polynom som helst i variabeln x och med koefficienter i $\mathbb{Z}$. Så exempelvis har vi

$$(x + 4) \cdot x^2 + (x^{21} - 3) \cdot (x^2 - 1) = x^3 + 4x^2 + x^{23} - x^{21} - 3x^2 + 3 =$$

$$= x^{23} - x^{21} + x^3 + x^2 + 3 \in I$$

Som sagt är det inte något produktivt projekt att försöka rabbla upp alla element i ett ideal utan i så fall bättre att endast titta på idealets generatorer. Men tänk om idealet genereras av ett oändligt antal generatorer då, tänker den oroliga läsaren. Men det visar sig att denna oro kan stillas.

Sats 1 (Hilberts bassats)**.** *Alla ideal som är en delmängd av en polynomring är ändligt genererade, dvs $I = \langle p_1, p_2, ..., p_n \rangle$ för något $n \in \mathbb{Z}$ för alla ideal $I \subset K[\boldsymbol{x}]$.*

Beviset för denna sats ligger utanför omfånget av denna uppsats och lämnas därför därhän. För intresserade läsare kan beviset återfinnas här: KÄLLA.

En bättre riktad oro är gentemot frågan om hur vi kan ta reda på huruvida ett givet element ligger i ett ideal eller inte. I exempel 3.2 tittar vi på ett ideal som genereras av tämligen enkla polynom, x^2 och $x^2 - 1$ och det visar sig att det ganska långa polynomet $= x^{23} - x^{21} + x^3 + x^2 + 3$ ligger i detta ideal. Det

är inte alls uppenbart att detta polynom ligger i idealet givet dess generatorer så hur ska man göra för att avgöra om ett element ligger i idealet? Om I endast genereras av ett enda element är detta inget avancerat projekt.

Exempel 3.3. *Antag att $I = \langle 3 \rangle$ är ett ideal över ringen $\mathbb{Z}$ och vi undrar om $97 \in I$. Det enda vi behöver göra är att se om det existerar något $n \in \mathbb{Z}$ så att $97 = 3n$. Vi kan enkelt konstatera att inget sådant n existerar efterom 97 inte är delbart med 3. På samma sätt om $I = \langle 3 \rangle$ är ett ideal över polynomringen $\mathbb{Q}[x]$ och vi undrar om $42x^{17} \in I$ behöver vi endast se om det existerar något polynom $p \in \mathbb{Q}[x]$ så att $3p = 42x^{17}$. Eftersom $\frac{42x^{17}}{3} = 14x^{17} \in \mathbb{Q}[x]$ har vi att $42x^{17} \in I$.*

Situationen blir dock genast mer komplicerad om I är ett ideal över en polynomring och dessutom genereras av mer än ett element, för att inte tala om hur komplicerad situationen blir ifall I är ett ideal över en polynomring i mer än en variabel. I nästa avnitt presenteras algoritmer för hur detta åtagande ska gå till men innan dess behöver vi säga något om varitet.

Definition 3.4 (Varitet)**.** *Låt I vara ett ideal med $I \subset K[\boldsymbol{x}]$. Mängden av punkter som löser alla polynom i I kallas variteten av I och betecknas $V(I)$: $V(I) = \{(a_1, a_2, ..., a_n) : a_i \in \overline{K}, p(a_1, a_2, ..., a_n) = 0, \forall p \in I\}$ där $\overline{K}$ är den minsta utvidgningen av K som är algebraiskt sluten.*

Definition 3.5 (Algebraisk slutenhet)**.** *Låt $K[\boldsymbol{x}]$ vara en polynomring över kroppen K. K kallas algebraiskt sluten om det för alla $p \in K[\boldsymbol{x}]$ gäller att $p(a) = 0 \Rightarrow a \in K$.*

Notera att definition 3.4 inte kräver att K är algebraiskt sluten. Snarare tvärtom, hävdar definition 3.4 att polynomen i $K[x]$ kan utvärderas i sådant som inte ligger i K.

Om $I = \langle p_1, p_2, ..., p_n \rangle$ är ett ideal över någon polynomring R kan variteten av I, $V(I)$ betraktas som mängden av lösningar till ekvationssystemet nedan:

$$\begin{cases} p_1 = 0 \\ p_2 = 0 \\ \vdots \\ p_n = 0 \end{cases}$$

Senare kommer vi att se hur detta begrepp är relevant för oss men för nu lägger vi detta åt sidan och tar en titt på Gröbnerbaser.

3.2 Gröbnerbaser

Givet ett polynom f och ett ideal I över någon polynomring $K[\boldsymbol{x}]$ vill vi hitta ett sätt att avgöra om $f \in I$. Beroende på hur idealet genereras och vilken ring I är ett ideal över görs detta på olika sätt. Från enklast till svårast blir frågorna: Hur avgör vi om $f \in I$ om

(i) $I = \langle p \rangle$ där $p \in K[x]$?
(ii) $I = \langle p_1, p_2, ..., p_n \rangle$ där $p_i \in K[x]$ för $i = 1, 2, ..., n$?
(iii) $I = \langle p \rangle$ där $p \in K[\boldsymbol{x}]$ med $\boldsymbol{x} = (x_1, x_2, ..., x_m)$?
(iv) $I = \langle p_1, p_2, ..., p_n \rangle$ där $p_i \in K[\boldsymbol{x}]$ för $i = 1, 2, ..., n$ med $\boldsymbol{x} = (x_1, x_2, ..., x_m)$?

I exempel 3.3 använder vi det faktum att alla multiplar av idealets generator också ligger i idealet. Samma princip gäller då vi har situationen (i). För alla heltal $a, b \in \mathbb{Z}$ gäller att det existerar två unika heltal $q, r \in \mathbb{Z}$ så att $a = qb + r$ med $0 \leq r < b$. Om $I = \langle b \rangle$ har vi att $a \in I$ om $r = 0$.

Analogt har vi för alla polynom $f, p \in K[x], p \neq 0$ att det existerar unika polynom $q, r \in K[x]$ så att $f = qp + r$ med $deg(r) < deg(p)$ eller $r = 0$. Om $I = \langle p \rangle$ har vi att $f \in I$ om $r = 0$. Ett annat sätt att skriva detta är $f \xrightarrow{p} r$ vilket läses "f kan reduceras med p till r". Denna notation kommer att användas flitigt i det som följer.

Exempel 3.4. *Låt $I = \langle x - 3 \rangle$ vara ett ideal över ringen $R = \mathbb{Z}[x]$. Vi vill avgöra om $f_1 = x^3 + 2x^2 - 14x - 3$ och $f_2 = x^4 - 4x^3 - 15x + 45$ ligger i idealet. Låt oss börja med f_1. För att hitta q_1, r_1 så att $f_1 = (x - 3)q_1 + r_1$ dividerar vi $f_1 = x^3 + 2x^2 - 14x - 3$ med $x - 3$. Polynomdivision ger:*

$$x^3 + 2x^2 - 14x - 3 = (x - 3)(x^2 + 5x + 1)$$

Se figur 3(a). Vi har att $f_1 \xrightarrow{x-3} 0$. Med andra ord är f_1 en multipel av $x - 3$ så $f_1 \in I$.

Vi gör på exakt samma sätt för att hitta q_2, r_2 så att $f_2 = (x - 3)q_2 + r_2$. Polynomdivision ger:

$$x^4 - 4x^3 - 15x + 45 = (x - 3)(x^3 - x^2 - 3x - 24) - 27$$

Se figur 3(b). Vi har att $f_1 \xrightarrow{x-3} -27 \neq 0$ så $f_2 \notin I$.

$$\begin{array}{r|l} x^3 + 2x^2 - 14x - 3 & x - 3 \\ -(x^3 - 3x^2) & \\ \hline 5x^2 - 14x - 3 & \\ -(5x^2 - 15x) & \\ \hline x - 3 & \\ -(x - 3) & \\ \hline 0 & \end{array} \qquad \begin{array}{r|l} x^4 - 4x^3 - 15x + 45 & x - 3 \\ -(x^4 - 3x^3) & \\ \hline -x^3 - 15x + 45 & \\ -(x^3 + 3x^2) & \\ \hline -3x^2 - 15x + 45 & \\ -(-3x^2 + 9x) & \\ \hline -24x + 45 & \\ -(-24x + 72) & \\ \hline -27 & \end{array}$$

(a) Polynomdivision mellan f_1 och $x - 3$

(b) Polynomdivision mellan f_2 och $x - 3$

Figur 3: Polynomdivisioner

Så för att avgöra om $f \in I$ om (i) $I = \langle p \rangle$ där $p \in K[x]$ använder vi bara polynomdivision mellan f och p. Om $f \xrightarrow{p} 0$ gäller $f \in I$, annars $f \notin I$. Låt oss nu titta på situation (ii) $I = \langle p_1, p_2, ..., p_n \rangle$ där $p_i \in K[x]$ för $i = 1, 2, ..., n$. Det visar sig att I kan beskrivas som $\langle sgd(p_1, ..., p_n) \rangle$ där $sgd(p_1, ..., p_n)$ syftar till den största gemensamma delaren till $p_1, ..., p_n$. Detta eftersom den största gemensamma delaren till ett polynom i en variabel kan uttryckas som en linjärkombination av polynomen i fråga. Så det visar sig att vi kan reducera problemet så att vi återigen har situation (i) $I = \langle p \rangle$ där $p = sgd(p_1, ..., p_n) \in K[x]$. Så för att avgöra om $f \in I$ om (ii) $I = \langle p_1, p_2, ..., p_n \rangle$ där $p_i \in K[x]$ för $i = 1, 2, ..., n$ använder vi polynomdivision mellan f och $sgd(p_1, ..., p_n)$. Om $f \xrightarrow{sgd(p_1,...,p_n)} 0$ gäller $f \in I$, annars $f \notin I$.

Situation (iii) $I = \langle p \rangle$ där $p \in K[\boldsymbol{x}]$ med $\boldsymbol{x} = (x_1, x_2, ..., x_m)$ blir dock mer komplicerad. Antag till exempel att $I = \langle p = x_1^2 + x_2 \rangle$ är ett ideal över $\mathbb{Q}[x_1, x_2]$ och vi vill veta om $f = x_1 x_2^2$ ligger i idealet. Vi hade önskat att kunna göra någon slags polynomdivision mellan f och p men polynomdivision är bara möjligt om dividenden f har högre grad än divisorn p. Men det är inte helt uppenbart om $deg(f) > deg(p)$. Man kan argumentera på följande sätt: "I den ledande termen i p har x_1 högre exponent än x_1 har i den ledande termen i f så $deg(p) > deg(f)$". Ett annat sätt att se på saken är följande argument: "Den ledande termen i f har *totalt* högre grad än den ledande termen i p så $deg(f) > deg(p)$. Båda argumenten verkar mer eller mindre rimliga men ger olika utslag på vilken av polynomen som har högst grad. Med anledning av detta behöver vi införa något slags regelverk kring vilka polynom som är av högre grad än andra.

Definition 3.6 (Godtagbar monomordning)**.** *En relation $\succ$ mellan monom är en godtagbar monomordning om följande kriterier är uppfyllda*

(i) $\succ$ är en total ordning, dvs att det för alla monom m_1, m_2 gäller att $m_1 \succ m_2$, $m_2 \succ m_1$ eller $m_1 = m_2$

(ii) $\succ$ är transitiv, dvs att det för alla monom m_1, m_2, m_3 gäller att om $m_1 \succ m_2$ och $m_2 \succ m_3$ så $m_1 \succ m_3$

(iii) $\succ$ är kompatibel med monommultiplikation, dvs att det för alla monom m, m_1, m_2 gäller att om $m_1 \succ m_2$ så $mm_1 \succ mm_2$

(iv) för alla monom $m \neq 1$ gäller $m \succ 1$

Antag att p är ett polynom. Med monomen till f menas termerna i f utan dess koefficient. Så om $f = 3x_1^3 x_4 + 21x_2$ har f monomen $x_1^3 x_4$ och x_2. Det finns många olika godtagbara monomordningar. Låt oss titta på två olika, deglex och den monomordning som vi kommer att använda, degrevlex.

Definition 3.7 (Deglex)**.** $m_1 = x_1^{\alpha_1} x_2^{\alpha_2} ... x_n^{\alpha_n} \succ m_2 = x_1^{\beta_1} x_2^{\beta_2} ... x_n^{\beta_n}$ *om $deg(m_1) > deg(m_2)$ eller om $deg(m_1) = deg(m_2)$ och vi för något k har $\alpha_1 = \beta_1, \alpha_2 = \beta_2, ..., \alpha_k > \beta_k$, där $deg(m)$ syftar till summan av exponenterna till samtliga variabler i m.*

Definition 3.8 (Degrevlex)**.** $m_1 = x_1^{\alpha_1} x_2^{\alpha_2} ... x_n^{\alpha_n} \succ m_2 = x_1^{\beta_1} x_2^{\beta_2} ... x_n^{\beta_n}$ *om $deg(m_1) > deg(m_2)$ eller om $deg(m_1) = deg(m_2)$ och vi för något k har $\alpha_n = \beta_n, \alpha_{n-1} = \beta_{n-1}, ..., \alpha_k < \beta_k$.*

I ord kan deglex beskrivas som en monomordning som, givet två monom av samma *totala* ordning, värderar variabler med lägre index högre. Så exempelvis har vi $x_1 \succ x_2$ då vi använder deglex eftersom x_1 har lägre index än x_2. Vi har $x_1 \succ x_2$ även då vi använder degrevlex men av något andra skäl. Degrevlex kan beskrivas som en monomordning som, givet två monom av samma *totala* ordning, värderar monom som i största möjliga mån "undviker" variabler med högre index högre. Så här har vi att $x_1 \succ x_2$ på grund av att x_1 till större grad undviker höga index än x_2. Låt oss titta på ett exempel där de två monomordningarna värderar olika för att göra skillnaden mer konkret.

Exempel 3.5. *Låt $m_1 = x_1x_4$ och $m_2 = x_2x_3$. Låt $\alpha_1, ..., \alpha_4$ beteckna exponenterna för $x_1, ...x_4$ i m_1 och analogt $\beta_1, ..., \beta_4$ beteckna exponenterna för $x_1, ..., x_4$ i m_2. För båda monomordningarna har vi $deg(m_1) = deg(m_2) = 2$.*

Med deglex har vi $m_1 \succ m_2$ eftersom $deg(m_1) = deg(m_2)$ och $\alpha_1 > \beta_1$. Med andra ord värderar deglex m_1 högre än m_2 på grund av förekomsten av x_1.

Med degrevlex har vi däremot $m_2 \succ m_1$ ty $deg(m_1) = deg(m_2)$ och $\beta_4 < \alpha_4$. Med andra ord värderar degrevlex m_2 högre än m_1 på grund av frånvaron av x_4.

Som en påminnelse till läsaren är den fråga vi söker besvara följande: Hur avgör vi om $f \in I$ om (iii) $I = \langle p \rangle$ där $p \in K[\boldsymbol{x}]$ med $\boldsymbol{x} = (x_1, x_2, ..., x_m)$? Vi söker alltså en algoritm som, givet $f, p \in K[\boldsymbol{x}]$ ger oss ett polynom r så att $f = qp + r$ för något polynom q så att $deg(r)$ minimeras. Denna algoritm skisseras nedan:

(1) (1) Låt $f, p \in K[\boldsymbol{x}]$ och $\succ$ vara någon godtagbar monomordning.

(2) Låt S vara mängden av termer i f som är delbara med $lt(p)$ där $lt(p)$ är den ledande termen i p.

(3) Om S är tom så är ingen reduktion möjlig och $r = f$.

(4) Om S är icke-tom, låt t vara termen i S med högst ordning med avseende på $\succ$. Genomför därefter reduktionen:

$$f^* = f - \frac{t}{lt(p)}p$$

(5) Upprepa från (2) med f^* istället för f så långt som möjligt.

För att besvara frågan genomför vi alltså algoritmen och om vi får $f \xrightarrow{p} 0$ har vi att $f \in I$, annars inte.

Exempel 3.6. *Låt $I \subset \mathbb{Q}[(x_1, x_2)]$ med $I = \langle 4 - x_2 \rangle$. Vi vill avgöra om följande två polynom ligger i idealet: $f_1 = 4x_1 + 3x_1x_2 - x_1x_2^2$, $f_2 = 4x_1x_2 - x_1x_2^2 + 3x_2 - 11$.*

Vi genomför algoritmen för $f_1 = 4x_1 + 3x_1x_2 - x_1x_2^2$ och $p = 4 - x_2$. För att underlätta för läsaren att följa med i algoritmen är stegen utmarkerade med

motsvarande siffra. (1) Vi använder monomordningen degrevlex. (2) Vi har $lt(p) = -x_2$ *och därmed* $S = \{3x_1x_2, -x_1x_2^2\}$. *(4) Eftersom* $S \neq \emptyset$ *och* $-x_1x_2^2 \succ 3x_1x_2$ *med avseende på degrevlex har vi* $t = -x_1x_2^2$. *Vi genomför reduktionen:*

$$f_1^* = f_1 - \frac{t}{lt(p)}p =$$

$$= 4x_1 + 3x_1x_2 - x_1x_2^2 - \frac{-x_1x_2^2}{-x_2}(4 - x_2) =$$
$$= 4x_1 + 3x_1x_2 - x_1x_2^2 - x_1x_2(4 - x_2) =$$
$$= 4x_1 + 3x_1x_2 - x_1x_2^2 - 4x_1x_2 + x_1x_2^2 =$$
$$= 4x_1 - x_1x_2$$

(5) Vi upprepar nu från (2) med f_1^*. *(2)* $S = \{-x_1x_2\}$. *(4) Eftersom* $S \neq \emptyset$ *har vi* $t = -x_1x_2$. *Vi genomför reduktionen:*

$$f_1^{**} = f_1^* - \frac{t}{lt(p)}p =$$

$$= 4x_1 - x_1x_2 - \frac{-x_1x_2}{-x_2}(4 - x_2) =$$
$$= 4x_1 - x_1x_2 - x_1(4 - x_2) =$$
$$= 4x_1 - x_1x_2 - 4x_1 + x_1x_2 = 0$$

(5) Vi upprepar nu från (2) med f_1^{**}. *(2)* $S = \emptyset$. *(3) Eftersom* $S = \emptyset$ *är ingen vidare reduktion är möjlig och* $r = f_1^{**} = 0$ *så* $f_1 \in I$.

Vi genomför algoritmen för $f_2 = 4x_1x_2 - x_1x_2^2 + 3x_2 - 11$ *och* $p = 4 - x_2$. *(1) Vi använder degrevlex. (2) Vi har* $lt(p) = -x_2$ *och därmed* $S = \{4x_1x_2, -x_1x_2^2, 3x_2\}$. *(4) Eftersom* $S \neq \emptyset$ *och* $-x_1x_2^2 \succ 4x_1x_2 \succ 3x_2$ *med avseende på degrevlex har vi* $t = -x_1x_2^2$. *Vi genomför reduktionen:*

$$f_2^* = f_2 - \frac{t}{lt(p)}p =$$

$$= 4x_1x_2 - x_1x_2^2 + 3x_2 - 11 - \frac{-x_1x_2^2}{-x_2}(4 - x_2) =$$
$$= 4x_1x_2 - x_1x_2^2 + 3x_2 - 11 - x_1x_2(4 - x_2) =$$
$$= 4x_1x_2 - x_1x_2^x + 3x_2 - 11 - 4x_1x_2 + x_1x_2^2 =$$
$$3x_2 - 11$$

(5) Vi upprepar nu från (2) med f_2^*. *(2)* $S = \{3x_2\}$. *(4) Eftersom* $S \neq \emptyset$ *har vi* $t = 3x_2$. *Vi genomför reduktionen:*

$$f_2^{**} = f_2^* - \frac{t}{lt(p)}p =$$

$$= 3x_2 - 11 - \frac{3x_2}{-x_2}(4 - x_2) =$$

$$= 3x_2 - 11 + 3(4 - x_2) = 1$$

(5) Vi upprepar nu från (2) med f_2^{**}. (2) $S = \emptyset$. (3) Eftersom $S = \emptyset$ är ingen vidare reduktion möjlig och $r = f_2^{**} = 1 \neq 0$ så $f_2 \notin I$.

Notera att resten enbart är unik sånär som på monomordning. Väljer vi en annan monomordning riskerar vi att få en annan rest. Som tur är har vi dock följande sats:

Sats 2. *Låt f, p vara två polynom så att $f = qp$ för något polynom q. Då gäller $f \xrightarrow{p} 0$ oavsett vilken monomordning vi väljer.*

Så enda fallet då vi säkert får en unik "rest" är då denna rest är lika med 0. Då återstår nu bara den fråga som är intressant för oss, hur avgör man om $f \in I$ om (iv) $I = \langle p_1, p_2, ..., p_n \rangle$ där $p_i \in K[\boldsymbol{x}]$ för $i = 1, 2, ..., n$ med $\boldsymbol{x} = (x_1, x_2, ..., x_m)$?

Vi söker nu en algoritm som, givet $f, p_1, p_2, ..., p_n \in K[\boldsymbol{x}]$ ger oss ett polynom r sådant att $f = q_1p_1 + q_2p_2 + ... + q_np_n + r$ för några polynom q_i så att $deg(r)$ minimeras. Denna algoritm är snarlik algoritmen som skisserades ovan med en liten skillnad.

(1) Låt $f, p_1, p_2, ..., p_n \in K[\boldsymbol{x}]$ och $\succ$ vara någon godtagbar monomordning.

(2) Låt S_i vara mängden av termer i f som är delbara med $lt(p_i)$ för $i = 1, 2, ..., n$

(3) Om alla S_i är tomma så är ingen reduktion möjlig och $r = f$

(4) Låt S_j vara den icke-tomma mängd med lägst index och låt t vara termen i S_j med högst ordning med avseende på $\succ$. Genomför därefter reduktionen:

$$f^* = f - \frac{t}{lt(p_j)}p_j$$

(5) Upprepa från (2) med f^* istället för f så långt som möjligt.

Precis som i situation (iii) genomför vi algoritmen för att avgöra om $f \in I$. Om $f \xrightarrow{p_1, p_2, ..., p_n} 0$ har vi att $f \in I$, annars inte.

Exempel 3.7. *Låt $I \subset \mathbb{Q}[(x_1, x_2)]$ med $I = \langle x_1x_2 + 1, x_1^2 + 1 \rangle$. Vi vill avgöra om följande polynom ligger i idealet: $f = x_1^2x_2 + x_1x_2^2 + x_1 + x_2$*

Vi genomför algoritmen för $f = x_1^1x_2 + x_1x_2^2 + x_1 + x_2$, $p_1 = x_1x_2 + 1$, $p_2 = x_1^2 + 1$. Återigen markeras stegen ut med motsvarande siffra för att underlätta läsningen. (1) Vi använder monomorningen degrevlex. (2) Vi har $lt(p_1) = x_1x_2$ och $lt(p_2) = x_1^2$ och därmed har vi $S_1 = \{x_1^2x_2, x_1x_2^2\}$ respektive $S_2 = \{x_1^2x_2\}$. (3) Eftersom $S_1, S_2 \neq \emptyset$ har vi (4) $S_j = S_1$ vilket ger $t = x_1^2x_2$. Vi genomför nu reduktionen:

$$f^* = f - \frac{t}{lt(p_1)}p_1 =$$

$$= x_1^2x_2 + x_1x_2^2 + x_1 + x_2 - \frac{x_1^2x_2}{x_1x_2}(x_1x_2 + 1) =$$
$$= x_1^2x_2 + x_1x_2^2 + x_1 + x_2 - x_1^2x_2 - x_1 =$$
$$x_1x_2^2 + x_2$$

(5) Vi upprepar nu från (2) med f^* istället för f. (2) $S_1 = \{x_1x_2^2\}$, $S_2 = \emptyset$. (3) Eftersom $S_1 \neq \emptyset$ har vi (4) $t = x_1x_2^2$. Vi genomför nu reduktionen:

$$f^{**} = f^* - \frac{t}{lt(p_1)}p_1 =$$

$$x_1x_2^2 + x_2 - \frac{x_1x_2^2}{x_1x_2}(x_1x_2 + 1) =$$
$$= x_1x_2^2 + x_2 - x_1x_2^2 - x_2 = 0$$

(5) Vi upprepar nu från (2) med f^{**} istället för f^*. (2) $S_1 = S_2 = \emptyset$. (3) Eftersom $S_1 = S_2 = \emptyset$ har vi att ingen vidare reduktion är möjlig och $f \xrightarrow{p_1,p_2} 0$ så $f \in I$.

Notera att det i exempel 3.7 inte finns någon särskild anledning till att vi sätter $p_1 = x_1x_2 + 1$ och $p_2 = x_1^2 + 1$. Eftersom $I = \langle x_1x_2 + 1, x_1^2 + 1\rangle$ och $I = \langle x_1^2 + 1, x_1x_2 + 1\rangle$ är exakt samma ideal kunde vi lika gärna ha haft $p_1 = x_1^2 + 1$ och $p_2 = x_1x_2 + 1$ istället.

Det verkar vid första anblick som att vi har hittat en metod för att avgöra om ett polynom ligger i ett ideal eller inte i situation (iv) men det finns ett dolt problem med denna metod. Problemet med detta är att resten vi får då vi reducerar f med $p_1, p_2, ..., p_n$ inte är unik. Som bekant har vi här att resten beror av monomordningen men problemet här är att resten också beror av reduktionsordningen. Tyvärr gäller här även att vi kan få $r = 0$ med en reduktionsordning och $r \neq 0$ med en annan så algoritmen kan ge falska negativa utslag.

Låt oss besöka exempel 3.7 igen men nu sätta $p_1 = x_1^2 + 1$ och $p_2 = x_1x_2 + 1$ istället.

Exempel 3.8. *Låt $I \subset \mathbb{Q}[(x_1, x_2)]$ med $I = \langle x_1^2 + 1, x_1x_2 + 1\rangle$. Vi vill avgöra om följande polynom ligger i idealet: $f = x_1^1x_2 + x_1x_2^2 + x_1 + x_2$*

Vi genomför algoritmen för $f = x_1^1x_2 + x_1x_2^2 + x_1 + x_2$, $p_1 = x_1^2 + x$, $p_2 = x_1x_2 + 1$. (1) Vi använder polynomordningen degrevlex. (2) Vi har $lt(p_1) = x_1^2$ och $lt(p_2) = x_1x_2$ och därmed har vi $S_1 = \{x_1^2x_2\}$ respektive $S_2 = \{x_1^2x_2, x_1x_2^2\}$. (3) Eftersom $S_1, S_2 \neq \emptyset$ har vi (4) $S_j = S_1$ och vilket ger $t = x_1^2x_2$. Vi genomför reduktionen:

$$f^* = f - \frac{t}{lt(p_1)}p_1 =$$

$$= x_1^2x_2 + x_1x_2^2 + x_1 + x_2 - \frac{x_1^2x_2}{x_1^2}(x_1^2 + 1) =$$
$$= x_1^2x_2 + x_1x_2^2 + x_1 + x_2 - x_1^2x_2 - x_2 =$$

$$= x_1x_2^2 + x_1$$

(5) Vi upprepar nu från (2) med f^* istället för f. (2) $S_1 = \emptyset$, $S_2 = \{x_1x_2^2\}$. (3) Eftersom $S_2 \neq \emptyset$ har vi (4) $S_j = S_2$ vilket ger $t = x_1x_2^2$. Vi genomför nu reduktionen:

$$f^{**} = f^* - \frac{t}{lt(p_2)}p_2 =$$

$$= x_1x_2^2 + x_1 - \frac{x_1x_2^2}{x_1x_2}(x_1x_2 + 1) =$$

$$= x_1x_2^2 + x_1 - x_1x_2^2 - x_2 =$$

$$x_1 - x_2$$

(5) Vi upprepar nu från (2) med f^{**} istället för f^*. (2) $S_1 = S_2 = \emptyset$. (3) Eftersom $S_1 = S_2 = \emptyset$ har vi att ingen vidare reduktion är möjlig och $f \xrightarrow{p_1,p_2} x_1 - x_2$ men vi såg i exempel 3.7 att $f \in I$.

Vi saknar något, och detta något är Gröbnerbaser.

Definition 3.9 (Gröbnerbas)**.** *Låt $K[x_1, .., x_n]$ vara en polynomring och låt I vara ett ideal till $K[x_1, ..., x_n]$. Vi definierar $l(I)$ som idealet som genereras av de ledande monomen hos elementen i I, dvs $l(I) = \langle lm(f) : f \in I\rangle$. En mängd $G = \{g_1, g_2, .., g_n\}$ som genererar I kallas en Gröbnerbas till I om $\langle lm(g_1), lm(g_2), ..., lm(g_n)\rangle = l(I)$*

En Gröbnerbas G är alltså en mängd flervariabelpolynom som genererar ett ideal I och det visar sig att Gröbnerbaser har följande fina egenskap. Om vi försöker reducera ett polynom f med polynomen i G är resten oberoende av reduktionsordningen. Så om vi, givet ett ideal I, kan hitta en Gröbnerbas till I så kan vi genomföra algoritmen ovan för att avgöra om $f \in I$ och vi kommer inte få några falska negativa utslag.

För att hitta en Gröbnerbas till ett givet ideal använder vi Buchbergeralgoritmen. Innan vi skisserar denna algoritm behöver vi definiera S-polynom.

Definition 3.10 (S-polynom)**.** *Låt p_i, p_j vara polynom. S-polynomet av p_i och p_j, $S_{i,j}$, definieras*

$$S_{i,j} = \frac{lcm(lt(p_i), lt(p_j))}{lt(p_i)}p_i - \frac{lcm(lt(p_i), lt(p_j))}{lt(p_j)}p_j$$

där $lcm(a, b)$ är den minsta gemensamma multipeln av a och b.

Eftersom $S_{i,j}$ är en linjärkombination av p_i och p_j har vi att om $p_i, p_j \in I$ så $S_{i,j} \in I$. Vi är nu redo att presentera Buchbergeralgoritmen.

(1) Låt $I = \langle p_i : p_i \in G\rangle$ där $G = \{p_1, p_2, ..., p_n\}$ och låt $\succ$ vara någon godtagbar monomordning.

(2) Låt $L = \{(1, 2), (1, 3), ...(1, n - 1), (1, n), (2, 3), (2, 4)..., (n - 1, n)\}$ vara mängden av heltalspar från 1 till n.

(3) Ta bort ett element (i, j) från L och beräkna $S_{i,j}$ för p_i, p_j. Reducera $S_{i,j}$ med elementen i G och låt $r_{i,j}$ beteckna resten.

(4a) Om $r_{i,j} = 0$ gör inget.

(4b) Om $r_{i,j} \neq 0$ och G innehåller n element, låt $p_{n+1} = r_{i,j}$ och lägg p_{n+1} i G. Lägg därefter till heltalsparen $(1, n+1), (2, n+1), ..., (n, n+1)$ i L.

(5a) Om $L \neq \emptyset$, upprepa från (3).

(5b) Om $L = \emptyset$ är G en gröbnerbas till I.

Vi återbesöker idealet i exempel 3.8 en sista gång.

Exempel 3.9. *Låt $I = \langle x_1^2 + 1, x_1x_2 + 1\rangle$. Vi vill hitta en Gröbnerbas till I.*

Vi genomför Buchbergers algoritm på polynomen som genererar I. Stegen markeras med motsvarande siffra. (1) Vi använder degrevlex och har $G = \{x_1^2 + 1, x_1x_2 + 1\}$. (2) $L = \{(1, 2)\}$. (3) Vi tar bort $(1, 2)$ från L, vilket ger $L = \emptyset$, och beräknar $S_{1,2}$ för p_1, p_2:

$$S_{1,2} = \frac{lcm(lt(p_1), lt(p_2))}{lt(p_1)}p_1 - \frac{lcm(lt(p_1), lt(p_2))}{lt(p_2)}p_2 =$$

$$= \frac{x_1^2x_2}{x_1^2}(x_1^2 + 1) - \frac{x_1^2x_2}{x_1x_2}(x_1x_2 + 1) =$$

$$= x_1^2x_2 + x_2 - x_1^2x_2 - x_1 =$$

$$= x_2 - x_1$$

Vi reducerar $S_{1,2}$ med p_1, p_2. Vi får att ingen reduktion är möjlig och $r_{1,2} = x_2 - x_1$. (4b) Vi lägger till $r_{1,2}$ i G och får $G = \{x_1^2 + 1, x_1x_2 + 1, x_2 - x_1\}$ respektive $L = \{(1, 3), (2, 3)\}$. (5a) Eftersom $L \neq \emptyset$ (3) tar vi bort $(1, 3)$ från L vilket ger $L = \{(2, 3)\}$ och beräknar $S_{1,3}$ för p_1, p_3:

$$S_{1,3} = \frac{lcm(x_1^2, -x_1)}{x_1^2}(x_1^2 + 1) - \frac{lcm(x_1^2, -x_1)}{-x_1}(x_2 - x_1) =$$

$$= \frac{x_1^2}{x_1^2}(x_1^2 + 1) - \frac{x_1^2}{-x_1}(x_2 - x_1) =$$

$$= x_1^2 + 1 + x_1x_2 - x_1^2 =$$

$$= x_1x_2 + 1$$

Vi reducerar $S_{1,3}$ med p_1, p_2, p_3. Vi får att (4a) $r_{1,3} = 0$ och lägger därför inte till något i G. (5a) Eftersom $L \neq \emptyset$ (3) tar vi bort $(2, 3)$ från L, vilket ger $L = \emptyset$, och beräknar $S_{2,3}$ för p_2, p_3:

$$S_{2,3} = \frac{x_1x_2, x_1}{x_1x_2}(x_1x_2 + 1) - \frac{x_1x_2, x_1}{-x_1}(x_2 - x_1) =$$

$$= x_1x_2 + 1 + x_2(x_2 - x_1) =$$
$$= x_1x_2 + 1 + x_2^2 - x_1x_2 =$$
$$= x_2^2 + 1$$

Vi reducerar $S_{2,3}$ *med* p_1, p_2, p_3. *Vi får att ingen vidare reduktion är möjlig och* $r_{2,3} = x_2^2 + 1$. *(4b) Vi lägger till* $r_{2,3}$ *i* G *och får* $G = \{x_1^2 + 1, x_1x_2 + 1, x_2 - x_1, x_2^2 + 1\}$ *respektive* $L = \{(1,4),(2,4),(3,4)\}$. *(5a) Eftersom* $L \neq \emptyset$ *(3) tar vi bort* $(1,4)$ *från* L, *vilket ger* $L = \{(2,4),(3,4)\}$, *och beräknar* $S_{1,4}$ *för* p_1, p_4*:*

$$S_{1,4} = \frac{x_1^2x_2^2}{x_1^2}(x_1^2 + 1) - \frac{x_1^2x_2^2}{x_2^2}(x_2^2 + 1) =$$
$$= x_1^2x_2^2 + x_2^2 - x_1^2x_2^2 - x_1^2 =$$
$$= x_2^2 - x_1^2$$

Vi reducerar $S_{1,4}$ *med* p_1, p_2, p_3, p_4 *och får (4a)* $r_{1,4} = 0$ *och lägger därför inte till något i* G. *(5a) Eftersom* $L \neq \emptyset$ *(3) tar vi bort* $(2,4)$ *från* L, *vilket ger* $L = \{(3,4)\}$, *och beräknar* $S_{2,4}$ *för* p_2, p_4*:*

$$S_{2,4} = \frac{x_1x_2^2}{x_1x_2}(x_1x_2 + 1) - \frac{x_1x_2^2}{x_2^2}(x_2^2 + 1) =$$
$$= x_1x_2^2 + x_2 - x_1^2x_2 - x_1 =$$
$$= x_2 - x_1$$

Vi reducerar $S_{2,4}$ *med* p_1, p_2, p_3, p_4 *och får (4a)* $r_{2,4} = 0$ *och lägger därför inte till något i* G. *(5a) Eftersom* $L \neq \emptyset$ *(3) tar vi bort (3,4) från* L, *vilket ger* $L = \emptyset$, *och beräknar* $S_{3,4}$ *för* p_3, p_4*:*

$$S_{3,4} = \frac{x_1x_2^2}{-x_1}(x_2 - x_1) - \frac{x_1x_2^2}{x_2^2}(x_2^2 + 1) =$$
$$= -x_2^2(x_2 - x_1) - x_1x_2^2 - x_1 =$$
$$= -x_2^3 + x_1x_2^2 - x_1x_2^2 - x_1 =$$
$$= -x_2^3 - x_1$$

Vi reducerar $S_{3,4}$ *med* p_1, p_2, p_3, p_4 *och får (5a)* $r_{3,4} = 0$ *och lägger därför inte till något i* G. *(5b) Eftersom* $L = \emptyset$ *är vi klara och* $G = \{x_1^2 + 1, x_1x_2 + 1, x_2 - x_1, x_2^2 + 1\}$ *är en Gröbnerbas till* I.

I exemplet ovan genereras I av två polynom men trots detta får vi en extremt lång beräkningsprocess för att hitta en Gröbnerbas. Senare kommer vi att jobba med ett ideal som genereras av 256 polynom. Visserligen kommer vi att kunna göra vissa förenklingar i beräkningarna tack vare att det är ett ideal över $\mathbb{Z}_2[\boldsymbol{x}]$ men det krävs fortfranade en stor mängd beräkningskraft för att hitta en Gröbnerbas till det gällande idealet. Detta är inte en beräkning som kommer att kunna göras för hand utan något som måste göras med dator. Se appendix för kod.

Sats 3 (Svag Nullstellensatz [2, Theorem 2.3.8])**.** *Under förutsättning att* $x_i^p - x_i \in I$ *för* $i = 1, 2, ..., n$ *så gäller* $V(I) = \emptyset$ *omm* $1 \in I$ *där* $I \subset \mathbb{Z}_p[x_1, x_2, ..., x_n]$

4 Modellering

Det finns flera olika sätt att modellera reglerna för shidoku men i det som följer kommer vi bara ägna oss åt modellering i $\mathbb{Z}_2[\boldsymbol{x}]$. Modelleringen följer till viss del den booleanska modelleringen av shidoku i Arnold et al. (2010) [1] med vissa modifikationer eftersom vi arbetar i polynomringen $\mathbb{Z}_2[\boldsymbol{x}]$.

4.1 Modellering av shidoku i $\mathbb{Z}_2$

Till varje cell i shidokunätet inför vi variabeln $x_{m,n}$ där m refererar till cellens rad och n refererar till cellens kolumn enligt figur 4.

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

Figur 4: Shidokunät med cellernas variabler markerade

För varje $m = 1, 2, 3, 4$ och $n = 1, 2, 3, 4$ har vi:

$$x_{m,n} = \begin{cases} 1, & \text{eller} \\ 2, & \text{eller} \\ 3, & \text{eller} \\ 4 \end{cases}$$

Men eftersom vi vill modellera reglerna för shidoku i $\mathbb{Z}_2[\boldsymbol{x}]$ kan vi bara jobba med de två elementen 0 och 1 så för varje cell, $x_{m,n}$ inför vi fyra variabler $x_{m,n,1}$, $x_{m,n,2}$, $x_{m,n,3}$, $x_{m,n,4}$ för vilka gäller:

$$x_{m,n,i} = \begin{cases} 1, & \text{eller} \\ 0 \end{cases}$$

Mer specifikt har vi att $x_{m,n,i} = 1$ om $x_{m,n} = i$ och $x_{m,n,i} = 0$ annars. Detta kriterium kan modelleras med ekvationen nedan för varje $x_{m,n,i}$, totalt $4 \cdot 4 \cdot 4 = 64$ stycken:

$$x_{m,n,i}^2 - x_{m,n,i} = 0 \tag{1}$$

Dessa ekvationer kan vid första anblick verka redundanta eftersom de bara säger oss att vi för varje variabel $x_{m,n,i}$ har att $x_{m,n,i} = 0$ eller $x_{m,n,i} = 1$ vilket vi redan kan sluta oss till i kraft av att 0 och 1 är de enda elementen i $\mathbb{Z}_2$. Men vi

kommer se att dessa ekvationer faktiskt är helt nödvändiga för oss när det är dags att besvara frågeställningen.

Med dessa variabler på plats kan vi börja modellera reglerna för shidoku. Specifikt är det följande två regler som ska modelleras:

(i) Exakt en av siffrorna 1-4 ska förekomma i en och samma cell.

(ii) Var och en av siffrorna 1-4 ska förekomma exakt en gång i en och samma region.

(i) Exakt en siffra i varje cell

Denna regel kan delas upp i två delkriterier, nämligen att åtminstone en av siffrorna 1-4 ska förekomma i varje cell och att det inte ska förekomma mer än en siffra i någon cell. För att garantera förekomsten av åtminsone en siffra behöver vi för varje cell $x_{m,n}$, totalt $4 \cdot 4 = 16$ ekvationer:

$$\sum_{i=1}^{4} x_{m,n,i} - 1 = 0 \tag{2}$$

Denna ekvation garanterar att minst en av de fyra variablerna är lika med 1, och därmed att cellen fylls av minst en av siffrorna 1-4, och utesluter också att ett jämnt antal av variablerna är lika med 1.

Bevis. Antag att $2p$ av variablerna för någon cell $x_{m,n}$ är lika med 1, säg $x_{m,n,1} = x_{m,n,2} = ... = x_{m,n,2p}$, med $p \in \mathbb{N}$. Vi har då:

$$\sum_{i=1}^{4} x_{m,n,i} - 1 = 2p - 1 \equiv_2 -1 \equiv_2 1 \neq 0$$

□

Men (2) utesluter inte att ett udda antal av de fyra variablerna är lika med 1. Det är inga problem med att en variabel kan vara lika med 1, snarare tvärtom vill vi att en av variablerna ska kunna vara lika med 1, men vi måste utesluta att tre av variablerna kan vara lika med 1 och det gör som sagt inte (2). Beakta följande. Antag att $x_{m,n,i_1} = x_{m,n,i_2} = x_{m,n,i_3} = 1$ för några $i_k \in \{1, 2, 3, 4\}$ och någon cell $x_{m,n}$. Vi har då:

$$\sum_{i=1}^{4} x_{m,n,i} - 1 = 1 + 1 + 1 - 1 = 2 \equiv_2 0$$

För att utesluta möjligheten att fler än en av variablerna är lika med 1 inför vi för varje cell $x_{m,n}$, totalt $4 \cdot 4 = 16$ stycken, följande ekvation:

$$\sum_{i=1, j=i+1}^{3} x_{m,n,i} x_{m,n,j} = 0 \tag{3}$$

Notera att produkten av varje parkombination av variablerna förekommer exakt en gång i summan. Antag nu, som ovan, att $x_{m,n,i_1} = x_{m,n,i_2} = x_{m,n,i_3} = 1$ för någon cell $x_{m,n}$. Av konstruktion kommer precis $\binom{3}{2}$ av summans termer vara lika med 1. Så vi får:

$$\sum_{i=1,j=i+1}^{3} x_{m,n,i}x_{m,n,j} = 1 \cdot \binom{3}{2} = 3 \equiv_2 1 \neq 0$$

Så (3) utesluter att tre av variablerna är lika med 1. Tillsammans garanterar (2) och (3) att exakt en av $x_{m,n,1}$, $x_{m,n,2}$, $x_{m,n,3}$, $x_{m,n,4}$ är lika med 1 för varje cell $x_{m,n}$.

(ii) Exakt en förekomst av varje siffra i varje region

Eftersom (2) och (3) garanterar att varje cell fylls av exakt en siffra räcker det att utesluta att samma siffra förekommer två gånger i samma region för att garantera att varje region innehåller alla siffror 1-4 exakt en gång.

Så för varje par av celler, $x_{m,n}$ och $x_{k,l}$, som ligger i samma region inför vi de fyra ekvationerna:

$$\begin{cases} x_{m,n,1}x_{k,l,1} = 0 \\ x_{m,n,2}x_{k,l,2} = 0 \\ x_{m,n,3}x_{k,l,3} = 0 \\ x_{m,n,4}x_{k,l,4} = 0 \end{cases} \tag{4}$$

Dessa fyra ekvationer utesluter att två celler i samma region innehåller samma siffra. Frågan är bara hur många sådana par som existerar i ett shidoku.

För varje rad existerar $\binom{4}{2} = 6$ och för varje kolumn existerar $\binom{4}{2} = 6$ sådana par. De par som återstår är de par av celler som ligger i samma box, men som inte ligger i samma rad eller kolumn. För varje box existerar också $\binom{4}{2} = 6$ sådana par men eftersom vi redan räknat de par som ligger i samma rad respektive kolumn måste vi räkna bort dessa. För varje box har vi redan räknat 2 par som ligger på samma rad och 2 par som ligger i samma kolumn så i varje rad finns $\binom{4}{2} - 4 = 2$ par som inte redan räknats. Eftersom vi har 4 rader, 4 kolumner respektive 4 boxar får vi totalt $4(6+6+2) = 56$ sådana par, så totalt har vi 56 uppsättningar av ekvation (4).

Med ekvationerna (1), (2), (3) och (4) modellerar vi precis reglerna för ett shidoku. Vi har 64 uppsättningar av (1), dvs 64 ekvationer, 16 uppsättningar av (2), dvs 16 ekvationer, 16 uppsättningar av (3), dvs 16 ekvationer, och 56 uppsättningar av (4), dvs 224 ekvationer. Totalt består vårat ekvationssystem alltså av $64 + 16 + 16 + 224 = 320$ ekvationer.

4.2 Modellering av sudoku i $\mathbb{Z}_2$

Med modelleringen av shidoku på plats är det inget jättekliv att anta att en analog modellering för sudoku skulle kunna vara lämplig. Precis som med shidoku namnger vi cellerna i sudokunätet så att varje cell tillskrivs en variabel

$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$	$x_{1,5}$	$x_{1,6}$	$x_{1,7}$	$x_{1,8}$	$x_{1,9}$
$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$	$x_{2,5}$	$x_{2,6}$	$x_{2,7}$	$x_{2,8}$	$x_{2,9}$
$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$	$x_{3,5}$	$x_{3,6}$	$x_{3,7}$	$x_{3,8}$	$x_{3,9}$
$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$	$x_{4,5}$	$x_{4,6}$	$x_{4,7}$	$x_{4,8}$	$x_{4,9}$
$x_{5,1}$	$x_{5,2}$	$x_{5,3}$	$x_{5,4}$	$x_{5,5}$	$x_{5,6}$	$x_{5,7}$	$x_{5,8}$	$x_{5,9}$
$x_{6,1}$	$x_{6,2}$	$x_{6,3}$	$x_{6,4}$	$x_{6,5}$	$x_{6,6}$	$x_{6,7}$	$x_{6,8}$	$x_{6,9}$
$x_{7,1}$	$x_{7,2}$	$x_{7,3}$	$x_{7,4}$	$x_{7,5}$	$x_{7,6}$	$x_{7,7}$	$x_{7,8}$	$x_{7,9}$
$x_{8,1}$	$x_{8,2}$	$x_{8,3}$	$x_{8,4}$	$x_{8,5}$	$x_{8,6}$	$x_{8,7}$	$x_{8,8}$	$x_{8,9}$
$x_{9,1}$	$x_{9,2}$	$x_{9,3}$	$x_{9,4}$	$x_{9,5}$	$x_{9,6}$	$x_{9,7}$	$x_{9,8}$	$x_{9,9}$

Figur 5: Sudokunät med cellernas variabler markerade

$x_{m,n}$ där m avser raden cellen ligger i och n avser cellens kolumn enligt figur 5. Låt oss nu testa att extrapolera ekvationerna ovan till ett system med många fler variabler. Istället för fyra variabler per cell inför vi nio variabler för varje cell, $x_{m,n,i}$ för $i = 1, 2, ..., 9$. Vi har då totalt $9^3 = 729$ variabler för vilka gäller:

$$x_{m,n,i} = \begin{cases} 1, & \text{om } x_{m,n} = i \\ 0, & \text{annars} \end{cases}$$

Vilket modelleras, precis som i fallet med shidoku, med följande ekvation för varje $x_{m,n,i}$, totalt $9 \cdot 9 \cdot 9 = 729$:

$$x_{m,n,i}^2 - x_{m,n,i} - 1 = 0 \tag{5}$$

(i) Exakt en siffra i varje cell

Om vi rakt av använder samma ekvationer som i fallet med shidoku har vi för varje cell $x_{m,n}$, totalt $9 \cdot 9 = 81$ stycken:

$$\sum_{i=1}^{9} x_{m,n,i} - 1 = 0 \tag{6}$$

$$\sum_{i=1, j=i+1}^{8} x_{m,n,i} x_{m,n,j} = 0 \tag{7}$$

Precis som med shidoku garanterar (6) att det för ett udda antal av $i = 1, 2, ..., 9$ gäller $x_{m,n,i} = 1$ och (7) utesluter att tre av variablerna är lika med 1. Frågan är om detta är tillräckligt. Antag att fem av variablerna är lika med 1, t ex $x_{m,n,1} = x_{m,n,2} = x_{m,n,3} = x_{m,n,4} = x_{m,n,5} = 1$. Eftersom alla produkter av parkombinationer förekommer exakt en gång i summan (7) har vi att exakt $\binom{5}{2} = 10$ av termerna i (7) är lika med 1. Vi har då:

$$\sum_{i=1,j=i+1}^{8} x_{m,n,i}x_{m,n,j} = 1 \cdot \binom{5}{2} = 10 \equiv_2 0$$

Så (7) utesluter inte att fem av variablerna är lika med 1. På samma sätt utesluter inte (7) att alla nio variablerna är lika med 1 eftersom vi då har att precis $\binom{9}{2} = 36$ av termerna i (7) är lika med 1 så att vi får:

$$\sum_{i=1,j=i+1}^{8} x_{m,n,i}x_{m,n,j} = 1 \cdot \binom{9}{2} = 36 \equiv_2 0$$

Däremot utesluter (7) att sju av variablerna är lika med 1. Detta eftersom vi i sådant fall har att $\binom{7}{2} = 21$ av termerna i (7) är lika med 1 så att vi får:

$$\sum_{i=1,j=i+1}^{8} x_{m,n,i}x_{m,n,j} = 1 \cdot \binom{7}{2} = 21 \equiv_2 1 \neq 0$$

Vi kan utesluta att fem av variablerna är lika med 1 genom att införa följande ekvation för varje cell $x_{m,n}$, totalt $9 \cdot 9 = 81$ stycken:

$$\sum_{i=1,j=i+1,k=j+1,l=k+1}^{6} x_{m,n,i}x_{m,n,j}x_{m,n,k}x_{m,n,l} = 0 \tag{8}$$

Notera att (8) innehåller produkten av varje fyr-kombination av variablerna exakt en gång. Antag nu $x_{m,n,i_1} = x_{m,n,i_2} = x_{m,n,i_3} = x_{m,n,i_4} = x_{m,n,i_5} = 1$ med $i_k \in \{1, 2, ..., 9\}$. Då kommer exakt $\binom{5}{4} = 5$ av termerna i (8) vara lika med 1 så vi får:

$$\sum_{i=1,j=i+1,k=j+1,l=k+1}^{6} x_{m,n,i}x_{m,n,j}x_{m,n,k}x_{m,n,l} = 1 \cdot \binom{5}{4} = 5 \equiv_2 1 \neq 0$$

Så utesluter (8) att fem av variablerna är lika med 1. Vi behöver dock ytterligare en ekvation eftersom (8) inte utesluter att alla nio variabler är lika med 1 ty beakta följande. Antag att $x_{m,n,1} = x_{m,n,2} = ... = x_{m,n,9} = 1$. Vi har då att precis $\binom{9}{4} = 126$ av variablerna är lika med 1 så vi får:

$$\sum_{i=1,j=i+1,k=j+1,l=k+1}^{6} x_{m,n,i}x_{m,n,j}x_{m,n,k}x_{m,n,l} = 1 \cdot \binom{9}{4} = 126 \equiv_2 0$$

Så inte heller (8) utesluter att samtliga variabler för någon cell är lika med 1. Därför inför vi för varje cell $x_{m,n}$, totalt $9 \cdot 9 = 81$ stycken, följande ekvation:

$$\prod_{i=1}^{9} x_{m,n,i} = 0 \tag{9}$$

Tillsammans garanterar (6), (7), (8) och (9) att $x_{m,n,i} = 1$ för exakt ett $i = 1, 2, ..., 9$ för varje cell $x_{m,n}$.

(ii) Exakt en förekomst av varje siffra i varje region

På samma sätt som med shidoku räcker det att utesluta att samma siffra förekommer två gånger i samma region. Så för varje par av celler $x_{m,n}$ och $x_{k,l}$ som ligger i samma region gäller:

$$\begin{cases} x_{m,n,1}x_{k,l,1} = 0 \\ x_{m,n,2}x_{k,l,2} = 0 \\ x_{m,n,3}x_{k,l,3} = 0 \\ x_{m,n,4}x_{k,l,4} = 0 \\ x_{m,n,5}x_{k,l,5} = 0 \\ x_{m,n,6}x_{k,l,6} = 0 \\ x_{m,n,7}x_{k,l,7} = 0 \\ x_{m,n,8}x_{k,l,8} = 0 \\ x_{m,n,9}x_{k,l,9} = 0 \end{cases} \tag{10}$$

Hur många sådana par existerar i ett sudokunät då? För varje rad existerar $\binom{9}{2} = 36$ sådana par och likaså för varje kolumn. Resterande par är sådana som ligger i samma box men som inte ligger på samma rad eller i samma kolumn. I varje box finns $3 \cdot \binom{3}{2} = 3 \cdot 3 = 9$ celler som också ligger på samma rad och likaså 9 celler som också ligger i samma kolumn. Så i varje box finns $\binom{9}{2} - (9 + 9) = 36 - 18 = 18$ par som inte också ligger på samma rad eller i samma kolumn. Eftersom vi har 9 rader, 9 kolumner och 9 boxar har vi alltså totalt $9 \cdot (36 + 36 + 18) = 810$ sådana par.

Med ekvationerna (5), (6), (7), (8), (9) och (10) modellerar vi precis reglerna för sudoku. Vi har 729 uppsättningar av (5), dvs 729 ekvationer, 81 uppsättningar av (6), dvs 81 ekvationer, 81 uppsättningar av (7), dvs 81 ekvationer, 81 uppsättningar av (8), dvs 81 ekvationer, 81 uppsättningar av (9), dvs 81 ekvationer och 810 uppsättningar av (10), dvs $9 \cdot 810 = 7290$ ekvationer. Totalt består vårat ekvationssystem av $729 + 81 + 81 + 81 + 81 + 7290 = 8343$ ekvationer.

5 Antal lösningar

Vi har nu gjort mycket av det grundarbete som behövs för att besvara den ursprungliga frågeställningen. Som nämnt i sektion 3.2 kan variteten till ett

ideal $I = \langle p_1, p_2, ..., p_n \rangle$ betraktas som lösningsmängden till ekvationssystemet:

$$\begin{cases} p_1 = 0 \\ p_2 = 0 \\ \vdots \\ p_n = 0 \end{cases}$$

Och vi har just sett att shidoku och sudoku kan modelleras som ekvationssystem där varje lösning till ekvationssystemet motsvarar en lösning givet ett tomt shidoku- eller sudokunät. Så vad vi gör nu är att vi skapar idealen över $\mathbb{Z}_2[\boldsymbol{x}]$ för shidoku respektive sudoku och studerar dessa.

Så hur många lösningar finns det egentligen till ett sudoku? Den saknade pusselbiten för att besvara denna fråga är följande sats:

Sats 4 ([2, Theorem 3.2.4])**.** *Låt $R = \mathbb{Z}_2[x_1, x_2, ..., x_n]$ och $I = \langle x_1^2 - x_1, x_2^2 - x_2, ..., x_n^2 - x_n, p_1, p_2, ..., p_m \rangle$. Då gäller att $|V(I)|$ är precis lika med antalet monom i R som inte delas av $lm(g_i)$ för något $g_i \in G$ där G är en Gröbnerbas till I med avseende på någon godtagbar monomordning.*

Så om vi kan hitta en Gröbnerbas till de tänkta idealen och ta reda på hur många monom i $\mathbb{Z}_2[\boldsymbol{x}]$ som inte delas av något ledande monom i Gröbnerbasen så har vi vårat svar!

Precis som med modelleringen börjar vi här med att först titta på den mindre komplexa strukturen innan vi tar oss an den mer komplexa.

5.1 Antal lösningar till shidoku

Vi börjar med att skapa det önskade idealet I över $\mathbb{Z}_2[\boldsymbol{x}]$ där $\boldsymbol{x} = (x_{m,n,i})$ för $m, n, i = 1, 2, ..., 4$. Från ekvation (1) får vi de 64 polynomen:

$$x_{m,n,i}^2 - x_{m,n,i} \tag{11}$$

för $m, n, i = 1, 2, ..., 4$. Från ekvation (2) får vi de 16 polynomen:

$$\sum_{i=1}^{4} x_{m,n,i} - 1 \tag{12}$$

för $m, n = 1, 2, ..., 4$. Från ekvation (3) får vi de 16 polynomen:

$$\sum_{i=1, j=i+1}^{3} x_{m,n,i} x_{m,n,j} \tag{13}$$

för $m, n = 1, 2, ..., 4$. Från ekvation (4) får vi de 224 polynomen:

$$\begin{cases} x_{m,n,1} x_{k,l,1} \\ x_{m,n,2} x_{k,l,2} \\ x_{m,n,3} x_{k,l,3} \\ x_{m,n,4} x_{k,l,4} \end{cases} \tag{14}$$

för alla par av celler, $x_{m,n}$ och $x_{k,l}$, för vilka gäller $m = k = 1, 2, ..., 4$ och $n \neq l$, för alla par för vilka gäller $n = l = 1, 2, ..., 4$ och $m \neq k$, för alla par för vilka gäller $\lceil \frac{m}{3} \rceil = \lceil \frac{k}{3} \rceil$ och $m \neq k$ och $n \neq l$ respektive för alla par för vilka gäller $\lceil \frac{n}{3} \rceil = \lceil \frac{l}{3} \rceil$ och $m \neq k$ och $n \neq l$.

Alla dessa polynom lägger vi i I och får då ett ideal som genereras av 320 polynom. Notera att antalet lösningar till ekvationssystemet är precis lika med $|V(I)|$. Eftersom I är ett ideal över polynomringen $\mathbb{Z}_2[x_{m,n,i}]$ där $m, n, i = 1, 2, ..., 4$ och polynomen $x^2_{m,n,i} - x_{m,n,i}$ för $m, n, i = 1, 2, ..., 4$ ligger i vårt ideal kan vi använda sats 4 för att hitta $|V(I)|$. Vi behöver först hitta en gröbnerbas till I och därefter hitta antalet monom i $\mathbb{Z}_2[\boldsymbol{x}]$ som inte delas av $lm(g_k)$ för något $g_k \in G$, men detta är tyvärr lättare sagt än gjort.

Nu är ett bra läge att påminna läsaren om exempel 3.9 där vi tog fram en Gröbnerbas till ett ideal över en ring med endast två variabler, $\mathbb{Z}_2[x_1, x_2]$, som genererades av de två polynomen $x_1^2 + 1$ och $x_1x_2 + 1$. Denna till synes mycket enkla uppgift krävde en förvånansvärt lång beräkningsprocess. Nu har vi ett ideal över en ring med 64 variabler som genereras av 320 polynom. Detta är ingen process som går att göra för hand utan måste göras med dator. Se appendix för kod.[1]

Det visar sig att detta är krävande även för en dator och koden tar ett par minuter att exekvera men i slutändan får vi att det existerar 288 lösningar till ett tomt shidokunät, vilket stämmer överens med resultat större matematiker än jag själv hittat [1].

Vill vi istället titta på ett partiellt ifyllt shidokunät så är det bara att vi tillför ytterligare polynom till vårat system. Antag till exempel att vi vill veta hur många lösningar som existerar till nätet i figur 6.

		1	
	3		

Figur 6: Partiellt ifyllt shidokunät

Då lägger vi till de två polynomen $x_{1,3,1} - 1$ och $x_{2,2,3} - 1$ och kör koden med de två nya polynomen.

[1] För att beräkna antalet lösningar har jag fått stor hjälp av Samuel Lundqvist att skriva kod i programmet Macaulay2.

5.2 Antal lösningar till sudoku

Det önskade idealet för sudoku bildas analogt med hur det önskade idealet för shidoku bildades i sektion 5.1. Precis på samma sätt som för shidoku måste vi här också först hitta en Gröbnerbas till I och därefter hitta antalet monom i $\mathbb{Z}_2[\boldsymbol{x}]$ som inte delas av $lm(g_k)$ för något $g_k \in G$. När handledare Samuel Lundqvist tillfrågades om samma kod, modifierad för sudoku, hade kunnat ge ett resultat på hur många lösningar som finns till ett tomt sudokunät svarade han nekande. "Absolut inte. Kanske om vi väntar 100 år, och inte för att exekveringen hade tagit så lång tid, utan för att vi hade behövt vänta på mycket bättre datorer." Med det sagt finns det metoder att hitta Gröbnerbaser på ett mer beräkningssparsamt sätt men teorin som krävs för dessa ligger långt utanför omfånget av denna uppsats.

5.3 En kombinatorisk lösning

Det finns även andra sätt att hitta antalet lösningar till ett tomt shidokunät. Nedan följer ett sätt inspirerat av Taalman (2007) [4].

Låt oss betrakta ett tomt shidokunät och se hur många möjligheter som existerar för var och en av cellerna. Vi använder samma notation som ovan för cellnumreringen. För var och en av cellerna $x_{m,n}$ gäller:

$$x_{m,n} = \begin{cases} 1, \text{ eller} \\ 2, \text{ eller} \\ 3, \text{ eller} \\ 4 \end{cases}$$

Vi börjar med att betrakta box 1, dvs cellerna $x_{1,1}$, $x_{1,2}$, $x_{2,1}$ och $x_{2,2}$. Eftersom vi ännu inte placerat något alls i vårt sudokunät har vi att det finns finns 4 möjliga fyllningar för $x_{1,1}$. För $x_{1,2}$ gäller:

$$x_{1,2} \neq x_{1,1}$$

Så det finns 3 möjligheter för $x_{1,2}$. För $x_{2,1}$ gäller:

$$x_{2,1} \neq \begin{cases} x_{1,1} \\ x_{1,2} \end{cases}$$

Så det finns 2 möjligheter för $x_{2,1}$. För $x_{2,2}$ gäller:

$$x_{2,2} \neq \begin{cases} x_{1,1} \\ x_{1,2} \\ x_{2,1} \end{cases}$$

Så det finns endast en möjlighet för $x_{2,2}$. Låt oss nu titta på resten av rad 1. För $x_{1,3}$ gäller:

$$x_{1,3} = \begin{cases} x_{1,1} \\ x_{1,2} \end{cases}$$

Så det finns 2 möjligheter för $x_{1,3}$. För $x_{1,4}$ gäller:

$$x_{1,4} \neq \begin{cases} x_{1,1} \\ x_{1,2} \\ x_{1,3} \end{cases}$$

Så det finns endast en möjlighet för $x_{1,4}$. Låt oss nu titta på resten av kolumn 1. För $x_{1,3}$ gäller:

$$x_{1,3} \neq \begin{cases} x_{1,1} \\ x_{1,2} \end{cases}$$

Så det finns 2 möjligheter för $x_{1,3}$. Slutligen gäller för $x_{1,4}$:

$$x_{1,4} \neq \begin{cases} x_{1,1} \\ x_{1,2} \\ x_{1,3} \end{cases}$$

Så det finns endast en möjlighet för $x_{1,4}$. Låt oss kalla ett nät vars första box, första rad och första kolumn är ifyllda för ett ordnat shidokunät. Ett möjligt ordnat nät ges i figur 7. Vi har just konstaterat hur många möjligheter som finns

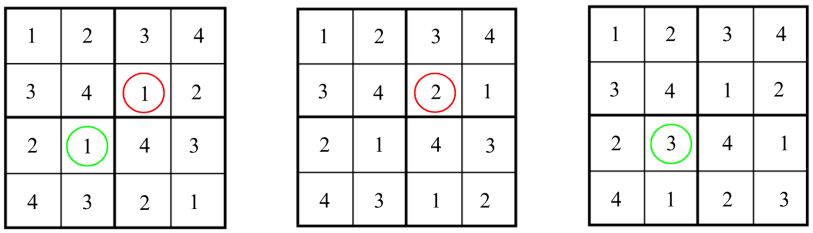
1	2	3	4
3	4		
2			
4			

Figur 7: Ett ordnat nät

för de relevanta cellerna i ett ordnat nät så totalt har vi $4 \cdot 3 \cdot 2 \cdot 1 \cdot 2 \cdot 1 \cdot 2 \cdot 1 = 96$ ordnade nät. Härifrån undersöker vi hur många lösningar som existerar givet ett visst ordnat nät genom att lösa shidokut rad för rad från vänster uppifrån. Eftersom det inte existerar en unik lösning till shidokut i figur 7 kommer vi för några celler behöva fatta ett beslut mellan två möjliga fyllningar. När ett sådant beslut fattas gör vi en kopia av nätet i vilken den gällande cellen fylls med den alternativa siffran. På så sätt genererar vi alla möjliga lösningar till det ordnade nätet. Givet det ordnade nätet i figur 7 får vi de tre lösningarna i figur 8.

Eftersom vi har 96 möjliga ordnade nät får vi totalt $96 \cdot 3 = 288$ möjliga shidokulösningar totalt.

Med denna metod har vi lyckats ta reda på hur många shidokulösningar som existerar på endast två sidor, så varför har vi ägnat så mycket arbete åt att



(a) Lösning 1 (b) Lösning 2 (c) Lösning 3

Figur 8: De tre möjliga lösningarna. Inringade siffror motsvarar de celler i vilka ett beslut mellan två möjliga fyllningar har fattats.

försöka bestämma detta antal med Gröbnerbaser? Jo, förhoppningen med att använda Gröbnerbaser var att extrapolera processen från den simpla strukturen i shidoku till det mer komplicerade systemet sudoku, och kanske till och med vidare till det ännu mer komplicerade fallet där nätet består av 16×16 celler.

Denna kombinatoriska lösningsmetod där vi med brute force tar reda på hur många lösningar som existerar fungerar utmärkt när vi tillämpar den på shidoku men inte fullt lika bra då vi tillämpar den på shidoku. Låt oss testa samma metod på sudoku. För sudoku existerar det $9! \cdot 6! \cdot 6! = 188116992000$ ordnade nät. Ett exempel på ett sådant nät ges i figur 9.

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

Figur 9: Ett ordnat sudokunät

Men hur ska vi bära oss åt nu? För cell $x_{2,4}$ har vi följande möjligheter:

$$x_{2,4} = \begin{cases} 1, \text{ eller} \\ 2, \text{ eller} \\ 3, \text{ eller} \\ 7, \text{ eller} \\ 8, \text{ eller} \\ 9 \end{cases}$$

Eftersom det finns sex olika möjliga fyllningar måste vi skapa sex kopior av nätet i figur 9. Men sen finns det fem möjligheter för $x_{2,5}$ så då måste vi göra fem kopior av alla sex kopior och på den vägen fortsätter det. Vi får väldigt snabbt en väldig massa sudokun att lösa för att ta reda på antalet lösningar.

Att metoden fungerar så bra för shidoku är på grund av en intressant egenskap som shidokunätet, men inte sudokunätet, besitter. Om vi betraktar en box i ett shidoku har vi två rader och två kolumner. Låt oss anta att vi vill placera siffran 1 i denna box. Om 1 inte kan vara i den första raden så *måste* 1 vara i den andra och vice versa. Detsamma gäller för kolumnerna. Om 1 inte kan vara i den första kolumnen så *måste* 1 vara i den andra och vice versa. För sudoku gäller inte detta. Om vi här betraktar en box har vi tre rader och tre kolumner. Om vi vill placera siffran 1 i någon given box och vet att 1 inte kan vara i den första raden så finns det nu två möjligheter vad gäller raden som 1 kan vara i. Vi kan placera ettan antingen i den andra raden *eller* i den tredje. Och detsamma gäller för kolumnerna! Där vi i fallet med shidoku endast har en möjlig placering av ettan i den givna boxen har vi här 8 möjliga placeringar. Komplexiteten ökar markant och det blir mycket svårare att hitta antalet lösningar med denna kombinatoriska brute force beräkning.

Om vi använder Gröbnerbaser så har vi en algoritmisk metod för att beräkna antalet lösningar till ett sudoku som fungerar i teorin, även om vi i praktiken behöver vänta i 100 år på svaret.

Referenser

[1] Arnold, E, Lucas S, Taalman L. (2010). Gröbner Basis Representations of Sudoku. *The college mathematics journal.* 10.4169/074683410X480203

[2] Gao, S. (2009). *Counting Zeros over Finite Fields Using Gröbner Bases.* [Masteruppsats]. Carnegie Mellon University.

[3] Sass, J. (2011). *Boolean polynomials and Gröbner bases: An algebraic approach to solving the SAT-problem.* [Masteruppsats]. Stockholms universitet.

[4] Taalman, L. (2007). Taking sudoku seriously. *Math Horizons, 15*(1), 5-9. http://www.jstor.org/stable/25678701

[5] Expressen, GT, Kvällsposten (vecka 30 2024). *Korsord*

Appendix

```
R=ZZ/2[x_{1,1,1}..x_{4,4,4}]
gens R

-- Polynom från ekvation (1)
Jfield=ideal (apply(gens R,i->i^2+i))

-- Polynom från ekvation (2)
Jcell=ideal 0_R;
for m from 1 to 4 do (
    for n from 1 to 4 do (
        Jcell=Jcell+((sum for i from 1 to 4 list x_{m,n,i}) + 1)
        );
    );

-- Polynom från ekvation (3)
Jcell'=ideal (0_R);
for m from 1 to 4 do (
    for n from 1 to 4 do (
        s = 0;
        for i from 1 to 3 do (
            for k from i+1 to 4 do (
                s = s + x_{m,n,i}*x_{m,n,k};
                );
            );
        Jcell'=Jcell'+ideal (s);
        );
    );

-- Polynom från ekvation (4) för radregionerna
Jrow = ideal (0_R);
for m from 1 to 4 do (
    for i from 1 to 4 do (
        for n from 1 to 3 do (
            for l from n+1 to 4 do (
                Jrow=Jrow+x_{m,n,i}*x_{m,l,i};
                );
            );
        );
    );
#Jrow_*-1

-- Polynom från ekvation (4) för kolumnregionerna
Jcol = ideal (0_R);
for l from 1 to 4 do (
```

```
    for i from 1 to 4 do (
        for m from 1 to 3 do (
            for k from m+1 to 4 do (
                Jcol=Jcol+x_{m,l,i}*x_{k,l,i};
                );
            );
        );
    );
#Jcol_*-1

-- Mall för polynomen från ekvation (4) för boxregionerna
idealFromBox = (i,j) -> (
    I = ideal (0_R);
    L = {};
    for k from i to i + 1 do (
        for h from j to j + 1 do (
            L = append(L,{k,h});
            );
        );

--Bildar parlistan av L
PL = subsets(L,2);
for p in PL do (
    for i from 1 to 4 do (
        I=I+x_(p#0 | {i}) * x_(p#1 | {i});
        );
    );
return I;
);

-- Polynom från ekvation (4) för boxregionerna
Jbox = idealFromBox(1,1) + idealFromBox(1,3) + idealFromBox(3,1) +
idealFromBox(3,3);

-- Idealet som består av samtliga polynom
I=Jfield + Jcell + Jcell' + Jrow + Jcol + Jbox;

-- Beräknar Gröbnerbasen (tar lite tid)
listGB= flatten entries gens gb I;

--Visar Gröbnerbasen faktoriserad
netList apply(listGB,factor)

--De ledande termerna i Gröbnerbasen (som är generatorer till LM(I))
lGB=ideal leadTerm ideal gens gb I;
```

```
--Här räknar vi antalet ledande termer i polynomringen som inte
ligger i LM(I)
sum for i from 0 to 10 list hilbertFunction(i,lGB)

--Här är ett försök att beräkna prod_i(f_i+1) + 1, men den terminerar
inte ens när vi gör det för den första gruppen av ekvationer.
S=R/Ifield;
KS = sub(Jcell,S) + sub(Jcell',S) + sub(Jrow,S) + sub(Jcol,S) +
sub(Ibox,S);

listGBS = flatten entries gens gb KS;

fJcell =1_S;
for i in ((sub(Jcell,S))_*) do (
    fJcell = fJcell * (i+1);
    print fJCell;
    );
```