Matrix Completion over GF(2) with Applications to Index Coding

Akhilesh Soni

Jeff Linderoth

Jim Luedtke

needs a haircut

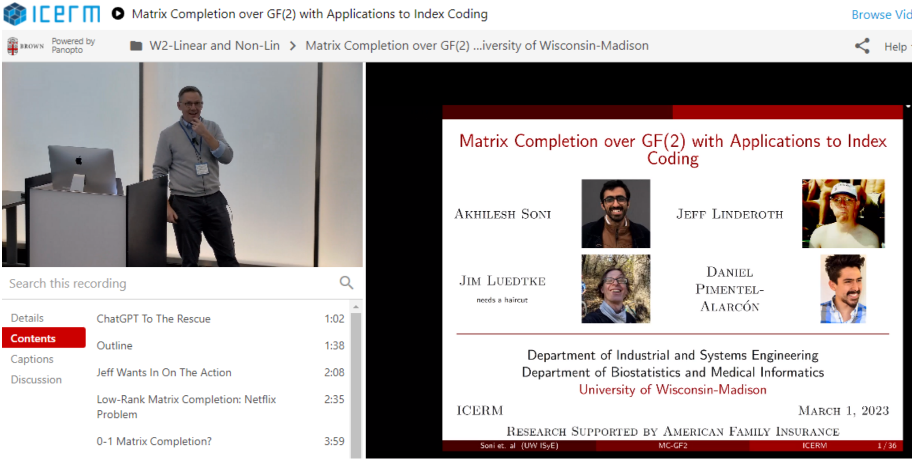
Daniel Pimentel-Alarcón

Department of Industrial and Systems Engineering
Department of Biostatistics and Medical Informatics
University of Wisconsin-Madison

MIP 2023 USC May 22, 2023

Research Supported by American Family Insurance

Apologies If You Were at ICERM



Apology Sonnet

To those who've witnessed my words' repetition,
I humbly kneel, seeking your forgiveness true.
For in this moment's time and its rendition,
I apologize for presenting the déjà vu.

Though echoes of past thoughts may fill the air,
And familiarity lingers in the room,
I strive to offer something fresh and rare,
To banish any sense of lingering gloom.

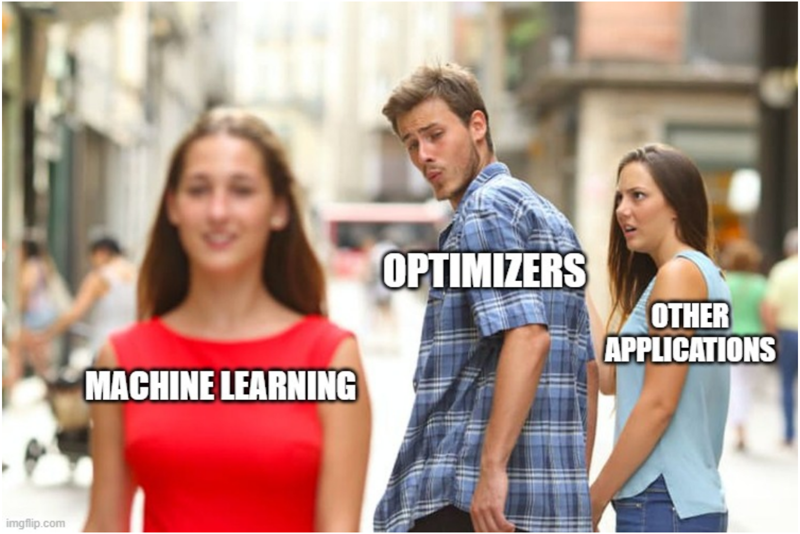
With newfound insight and renewed inspiration,
I promise to deliver a different voice,
To honor your time, your valued attention,
And grant you a reason to rejoice.

So, please accept my sincere apology,
As I endeavor to bring novelty.

Outline

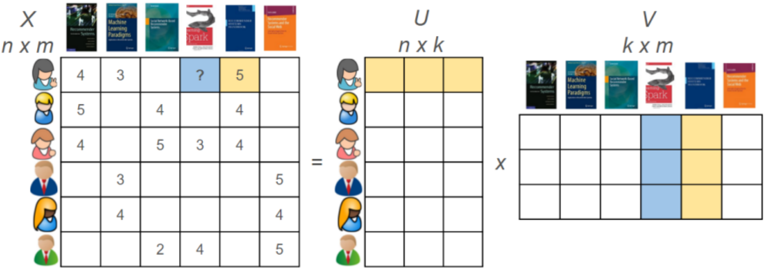
- Matrix completion
- Binary matrix factorization and completion
- Index coding
- Three IP Formulations
  1. McCormick + Integer Variable
  2. McCormick + Parity Disjunction
  3. McCormick-Free
- A Few New Results!
- Less than impressive computational results

Jeff Wants In On The Action



Low-Rank Matrix Completion: Netflix Problem

- There exists a matrix $X \in \mathbb{R}^{d \times n}$ whose entries are only known for a fraction of the elements $\Omega \subset [d] \times [n]$
- To complete the matrix, we must assume some structure.
- Here we assume X is low-rank: $X = UV$ for some $U \in \mathbb{R}^{d \times r}$, $V \in \mathbb{R}^{r \times n}$



0-1 Matrix Completion?

- In some earlier work sponsored by American Family, we did a combination of matrix completion and clustering—Subspace clustering with missing data
- They asked us to try it out on their data matrix—which was a 0-1 matrix (?!)

Well, Duh!?!

- Doing "normal" low-rank matrix completion methods in $\mathbb{R}$, are *not* going to give 0-1 values for the missing entries

What to do?

- Don't do it over $\mathbb{R}$.
- What about Boolean Algebra, Logical Or, $(1 + 1 = 1)$— natural for revealing "low-dimensional" characteristics

Boolean Algebra: $1+1 = 1$



$$
X = \begin{array}{r} \\ \text{Long Hair} \\ \text{Loves MIP} \\ \text{Cheesehead} \end{array}
\begin{array}{c} \begin{array}{ccc} \text{Simge} & \text{Jim} & \text{Jeff} \end{array} \\ \left[\begin{array}{ccc} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{array}\right] \end{array}
$$

Two Groups of People, Two Traits

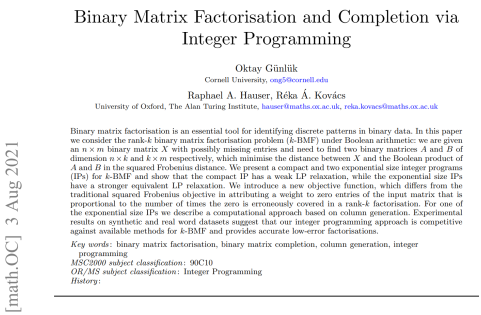
- Simge and Jim have long hair and love MIP
- Jim and Jeff love MIP and are cheeseheads

Two Factors

$$X = \begin{array}{r} \\ \text{Long Hair} \\ \text{Loves MIP} \\ \text{Cheesehead} \end{array} \begin{array}{c} \begin{array}{ccc} \text{Simge} & \text{Jim} & \text{Jeff} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{array} = \begin{array}{c} \begin{array}{cc} \text{T1} & \text{T2} \end{array} \\ \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \end{array} \circ \begin{array}{c} \begin{array}{ccc} \text{Simge} & \text{Jim} & \text{Jeff} \end{array} \\ \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \end{array}$$

- Writing $X = \bigvee_{k=1}^{r} u^k (v^k)^\top$ reveals the fundamental "traits", and classifies individuals depending on which traits they have
- So we started working on integer programming approaches to matrix factorization and completion in Boolean algebra

I Hate This Guy



Binary Matrix Factorisation and Completion via Integer Programming

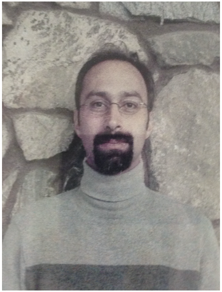
Oktay Günlük
Cornell University, ong5@cornell.edu

Raphael A. Hauser, Réka Á. Kovács
University of Oxford, The Alan Turing Institute, hauser@maths.ox.ac.uk, reka.kovacs@maths.ox.ac.uk

Binary matrix factorisation is an essential tool for identifying discrete patterns in binary data. In this paper we consider the rank-k binary matrix factorisation problem (k-BMF) under Boolean arithmetic: we are given an $n \times m$ binary matrix X with possibly missing entries and need to find two binary matrices A and B of dimension $n \times k$ and $k \times m$ respectively, which minimise the distance between X and the Boolean product of A and B in the squared Frobenius distance. We present a compact and two exponential size integer programs (IPs) for k-BMF and show that the compact IP has a weak LP relaxation, while the exponential size IPs have a stronger equivalent LP relaxation. We introduce a new objective function, which differs from the traditional squared Frobenius objective in attributing a weight to zero entries of the input matrix that is proportional to the number of times the zero is erroneously covered in a rank-k factorisation. For one of the exponential size IPs we describe a computational approach based on column generation. Experimental results on synthetic and real word datasets suggest that our integer programming approach is competitive against available methods for k-BMF and provides accurate low-error factorisations.

Key words: binary matrix factorisation, binary matrix completion, column generation, integer programming
MSC2000 subject classification: 90C10
OR/MS subject classification: Integer Programming
History:

[math.OC] 3 Aug 2021



Oktay Ruined It—Nothing Left To Do

- IP Formulations
- Strong Formulations
- Column Generation Approaches.

$$\mathbb{F}_2?$$

$$1 + 1 = 0$$

Binary Matrix Factorization/Completion

Matrix Factorization

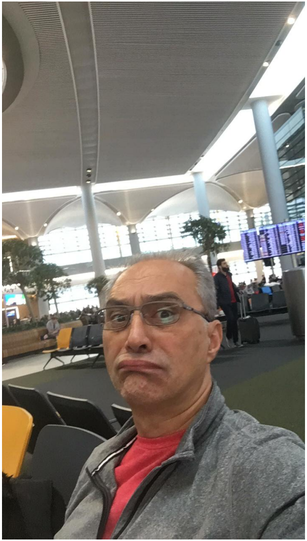
- Boolean: Find smallest r such that $X = \bigvee_{k=1}^{r} u^k (v^k)^\top$, where $u^k \in \{0,1\}^d, v^k \in \{0,1\}^n$. This is hard
- $\mathbb{F}_2$: Find smallest r such that $X = \bigoplus_{k=1}^{r} u^k (v^k)^\top$, where $u^k \in \{0,1\}^d, v^k \in \{0,1\}^n$. This is easy

---

Matrix Completion. Given $\Omega \subset [d] \times [n]$, $X_{ij} \in \{0,1\}$ $\forall ij \in \Omega$, $r \in \mathbb{Z}_+$

- Find $u^k \in \{0,1\}^d, v^k \in \{0,1\}^n$ to $\min \|X_{ij} - \bigvee_{k=1}^{r} u^k (v^k)^\top)\|_\Omega$. This is hard.
- Find $u^k \in \{0,1\}^d, v^k \in \{0,1\}^n$ to $\min \|X_{ij} - \bigoplus_{k=1}^{r} u^k (v^k)^\top)\|_\Omega$. This is hard.

An Honest To God Quotation.



"Matrix Completion in $\mathbb{F}_2$?!?!
Why on earth would anyone want
to solve that problem?"

Index Coding (with Side Information)

- We have a collection of n messages/packets, each in $\{0, 1\}^t$, and a collection of n receivers.
  - Each receiver wants to know one of the messages
  - Each receiver "knows" (has cached) some subset of the packets—Just not the one it wants to know
- Central broadcaster knows which packets are cached at each receiver

Index Coding

Broadcast a minimum number of messages so that each receiver can recover/compute its message using their local information

Intuition

Send a basis of "known" information $\Rightarrow$ each receiver can compute their own message. Min rank is minimum number of messages

Index Coding: Example

Receiver	Has Messages
1	2,5
2	1,5
3	2,4
4	2,3
5	1,3,4

$$X = \begin{array}{c} \\ M1 \\ M2 \\ M3 \\ M4 \\ M5 \end{array} \begin{array}{c} \begin{array}{ccccc} R1 & R2 & R3 & R4 & R5 \end{array} \\ \begin{bmatrix} 1 & - & 0 & 0 & - \\ - & 1 & - & - & 0 \\ 0 & 0 & 1 & - & - \\ 0 & 0 & - & 1 & - \\ - & - & 0 & 0 & 1 \end{bmatrix} \end{array}$$

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

- Broadcast two messages: $(M1 + M2 + M5, M2 + M3 + M4)$
- All receivers can reconstruct their desired message

Matrix Completion in $\mathbb{F}_2$?—State of the Art?

- No exact method in literature for matrix completion in $\mathbb{F}_2$ (!?)
- Heuristic pruning-based enumeration method in Esfahanizadeh, Lahuoti, and Hassibi, able to find (known) min rank solution for 7 by 7 instance every time in around 1 second.
- For 14 by 14 instance, in 30 min, they (sometimes) find rank 5 solution, sometimes find rank 6 solution.

MIP People Do It Exactly

Or at least up to floating point accuracy?

- We aim to build first(?) exact solver for this class of problems

Formulations for Matrix Completion in $\mathbb{F}_2$

- Some sets we will use

$$\mathcal{I} := \{(u, v, z) \in \{0,1\}^{2r+1} \mid z = \oplus_{k=1}^{r} u_k v_k\}$$
$$\mathcal{P} := \{(y, z) \in \{0,1\}^{r+1} \mid z = \oplus_{k=1}^{r} y_k\}$$
$$\mathcal{M} := \{(u, v, y) \in \{0,1\}^{3r} \mid y_k = u_k v_k \ \forall k \in [r]\}$$

- Note that $\mathrm{proj}_{u,v,z}(\mathcal{P} \cap \mathcal{M}) = \mathcal{I}$[1]
- Matrix Completion in $\mathbb{F}_2$:

$$\min \sum_{(ij)\in\Omega} |X_{ij} - z_{ij}|$$
$$(u^i, v^j, z_{ij}) \in \mathcal{I}_{ij} \ \forall ij \in \Omega$$

- Note that $u^i, v^j \in \{0,1\}^r$

[1] Notation Abuse!

Writing $\mathcal{M}$ as MIP

- Everyone (at least at this meeting) knows how to write $\mathcal{M}$ as the set of $\{0,1\}$-points inside a polyhedron. ($\mathcal{M}$ is for McCormick.)

$$\mathcal{M} = \{(u, v, y) \in \{0, 1\}^{3r} \mid y_k \leq u_k, y_k \leq v_k, y_k \geq u_k + v_k - 1 \ \forall k \in [r]\}$$

- Oktay told me that

$$\begin{aligned} \mathrm{LP}(\mathcal{M}) := \{(u, v, y) \in [0, 1]^{3r} \mid y_k \leq u_k, y_k \leq v_k \\ y_k \geq u_k + v_k - 1 \ \forall k \in [r]\} = \mathrm{conv}(\mathcal{M}) \end{aligned}$$

- It is also true (by separability) that

$$\mathrm{conv}(\mathcal{P} \cap \mathcal{M}) = \mathrm{conv}(\mathcal{P}) \cap \mathrm{conv}(\mathcal{M}).$$

Writing $\mathcal{P}$ as MIP

- Consider the general integer set:

$$\mathcal{Z} := \{(y, z, t) \in \{0, 1\}^{r+1} \times \mathbb{Z} \mid \sum_{k=1}^{r} y_k - 2t = z\}$$

- It is easy to see that $\mathcal{Z} = \mathcal{P}$
- So we have our "first" MILP formulation for matrix completion in $\mathbb{F}_2$:

$$\min \quad \sum_{(ij)\in\Omega} |X_{ij} - z_{ij}|$$

$$(u^i, v^j, y^{ij}) \in \mathcal{M}_{ij} \quad \forall ij \in \Omega$$
$$(y^{ij}, z_{ij}, t_{ij}) \in \mathcal{Z}_{ij} \quad \forall ij \in \Omega$$

Computational Experiments

- $X \in \{0,1\}^{10\times 10}$ will have $\mathbb{F}_2$-rank 4.
- Use MIP formulation to find "closest" rank r matrix for $r \leq 4$
- Let Ω be all matrix elements, and then start to (randomly) remove a fraction of the entries

Computational Results

% Missing	Rank	Time	Nodes	Opt
0	1	0.05	1	36
0	2	41.81	70237	24
0	3	7184.56	10437394	12
0	4	0.49	1	0
10	1	0.03	1	31
10	2	14.04	27757	17
10	3	320.59	996422	7
10	4	0.03	1	0
20	1	0.01	1	26
20	2	2.91	5872	14
20	3	4106.07	13393830	8
20	4	2.55	2430	0

Results are a Pig!

- 460 binary vars, 100 integer vars $>$ 10M nodes?

Improving The Pig

- The LP relaxation of the parity condition:

$$\mathrm{LP}(\mathcal{Z}) := \{(y, z, t) \in [0,1]^{r+1} \times \mathbb{R}_+ \mid 2t = \sum_{i=1}^{r} y_i - z\}$$

is very far from the convex hull of the true parity conditions:

$$\mathrm{proj}_{yz}\, \mathrm{LP}(\mathcal{Z}) \subset \mathrm{conv}(\mathcal{P})$$

- But lots is known about how to model parity conditions

Parity Polyhedra

$$P_E = \text{conv}\{x \in \{0,1\}^n \mid \sum_{i=1}^{n} x_i \text{ is even }\}$$

$$P_O = \text{conv}\{x \in \{0,1\}^n \mid \sum_{i=1}^{n} x_i \text{ is odd }\}$$

$$P_E = \{x \in [0,1]^n \mid \sum_{i \in S} x_i - \sum_{i \notin S} x_i \leq |S| - 1, \forall \text{ odd } S \subset [n]\}$$

$$P_O = \{x \in [0,1]^n \mid \sum_{i \in S} x_i - \sum_{i \notin S} x_i \leq |S| - 1, \forall \text{ even } S \subset [n]\}$$

- There are also small (even linear-size) extended formulations for P_E and P_O
- From these, and using disjunctive programming, we can give an extended formulation for $\text{conv}(\mathcal{P})$

One Extended Formulation for $\mathrm{conv}(\mathcal{P})$

- Let $D \in [0,1]^{3r+1}$ be the set of points satisfying bound constraints and the inequalities

$$
\begin{aligned}
&\sum_{k \in S} y_k^o - \sum_{k \notin S} y_k^o \leq (|S|-1)z && \forall \text{ even } S \subseteq [r] \\
&\sum_{k \in S} y_k^e - \sum_{k \notin S} y_k^e \leq (|S|-1)(1-z) && \forall \text{ odd } S \subseteq [r] \\
&y_k = y_k^o + y_k^e && \forall k \in [r] \\
&y_k^o \leq z && \forall k \in [r] \\
&y_k^e \leq 1 - z && \forall k \in [r]
\end{aligned}
$$

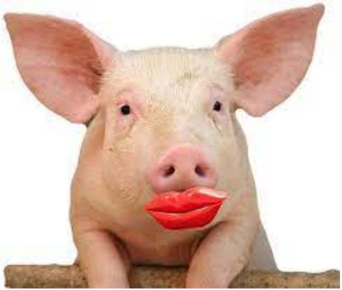
Thms:

$$\mathrm{conv}(\mathcal{P}) = \mathrm{proj}_{y,z} D \qquad \mathrm{conv}(\mathcal{P} \cap \mathcal{M}) = D \cap \mathrm{LP}(\mathcal{M}) = \mathrm{conv}(\mathcal{I})$$

MIP Formulation 2: LipStick on the Pig

$$\min \quad \sum_{(ij)\in\Omega} |X_{ij} - z_{ij}|$$

$$\begin{aligned} (u^i, v^j, y^{ij}) &\in \mathcal{M}_{ij} \quad \forall (ij) \in \Omega \\ (y^{ij}, y^{o,ij}, y^{e,ij}, z_{ij}) &\in D_{ij} \quad \forall (ij) \in \Omega \\ z_{ij} &\in \{0, 1\} \quad \forall ij \in \Omega \end{aligned}$$



MIP1 (Pig) v. MIP2 (Pig w/Lipstick)

MIP	% Missing	Rank	Time	Nodes	Opt
1	0	2	41.81	70237	24
2	0	2	9.42	13746	24
1	0	3	7184.56	10437394	12
2	0	3	2137.15	1272534	12
1	10	2	14.04	27757	17
2	10	2	6.63	20296	17
1	10	3	320.59	996422	7
2	10	3	357.02	353021	7
1	20	2	2.91	5872	14
2	20	2	3.64	8927	14
1	20	3	4106.07	13393830	8
2	20	3	2199.89	2366186	8

Team Reactions

"Why do you all keep talking about putting lipstick on a pig?"

"Aunque la mona se vista de seda, mona se queda"

(You can dress a monkey in silk, but it's still a monkey)

Keep Trying—Let's Get That Monkey

- Can we directly model the set

$$\mathcal{I} = \{(u, v, z) \in \{0, 1\}^{2r+1} \mid z = \oplus_{k=1}^{r} u_k v_k\}$$

without using auxiliary variables?

- Yes! Let $\mathfrak{T}$ be the set of all tri-partitions of $[r]$

$$\mathfrak{T} := \{S \subseteq [r], Q \subseteq [r], T \subseteq [r] \mid S \cup Q \cup T = [r] \\ S \cap Q = \emptyset, S \cap T = \emptyset, Q \cap T = \emptyset\}$$

- Consider families of inequalities

$$z + u(S) + v(S) - u(Q) - v(T) \leq 2|S| \quad \forall (S, Q, T) \in \mathfrak{T}, |S| \text{ even} \quad (1)$$
$$z - u(S) - v(S) + u(Q) + v(T) \geq 1 - 2|S| \quad \forall\ (S, Q, T) \in \mathfrak{T}, |S| \text{ odd} \quad (2)$$

Where Do They Come From?

- We found them via facet-hunting with PORTA, but they can be derived as follows:
- Choose an index $i \in [r]$ and create a tri-partition of $[r] \setminus i$, fixing

$$S := \{i \mid u_i = v_i = 1\}$$
$$Q := \{i \mid u_i = 0\}$$
$$T := \{i \mid v_i = 0\}$$

- If $|S|$ is even, then feasible points on face of $\mathcal{I}$ satisfy $z = u_i v_i \oplus 0$
- The inequality $z \geq u_i + v_i - 1$[2] is facet-defining for this face
- Lifting

$$u_i + v_i - z + \sum_{k \in S} \alpha_k (1 - u_k) + \sum_{k \in S} \beta_k (1 - v_k) + \sum_{k \in Q} \alpha_k u_k + \sum_{k \in T} \beta_k v_k \leq 1$$

Gives (2)

[2] Hello Dr. McCormick

Derivation, Continued

- If $|S|$ is odd, the feasible points on face of $\mathcal{I}$ satisfy $z = u_i v_i \oplus 1$
- The inequality $z \le 2 - u_i - v_i$ is facet-defining for this face
- Lifting

$$u_i + v_i + z + \sum_{k \in S} \alpha_k (1 - u_k) + \sum_{k \in S} \beta_k (1 - v_k) + \sum_{k \in Q} \alpha_k u_k + \sum_{k \in T} \beta_k v_k \le 2$$

  Gives (1)
- Can also get the inequalities (1) from (2) by the transformation $z \rightarrow 1 - z$.
  - When lifting, it suffices to consider the face with remainder (fixed) term 0.

Theorems

Theorem

- These (exponentially many in r) inequalities give a direct formulation of $\mathcal{I}$:

$$\mathcal{F} = \{(u, v, z) \in \{0, 1\}^{2r+1} \mid (1), (2)\}$$

- All inequalities are necessary

"Theorem" (from ICERM)

- The LP relaxation of the set is the convex hull

$$\mathrm{conv}(\mathcal{I}) = \{(u, v, z) \in [0, 1]^{2r+1} \mid (1), (2)\}$$

- "Theorem" because Jim hasn't proved it yet

"Theorem" No More!

- Akhilesh rose to the challenge, and proved the result, but it was more challenging than we expected.

Proof Mechanism

- For arbitrary objective function, construct an integer-valued feasible solution to the primal and a feasible solution to the dual of the same objective value.

$$\max_{(u,v,z)\in[0,1]^{2r+1}} \{c^\top u + d^\top v + fz \mid (1),(2)\} \tag{P}$$

Dual LP

$$\min \sum_{(S,Q,T)\in\mathfrak{T}} 2|S|\pi_{SQT} - \sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ |S| \text{ odd}}} \pi_{SQT} + \sum_{i=1}^{r} \mu_i + \sum_{i=1}^{r} \eta_i + \gamma \qquad \text{(D)}$$

$$\sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ |S| \text{ even}}} \pi_{SQT} - \sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ |S| \text{ odd}}} \pi_{SQT} + \gamma \geq f$$

$$\sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ S\ni i}} \pi_{SQT} - \sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ Q\ni i}} \pi_{SQT} + \mu_i \geq c_i \quad \forall i \in [r]$$

$$\sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ S\ni i}} \pi_{SQT} - \sum_{\substack{(S,Q,T)\in\mathfrak{T}:\\ T\ni i}} \pi_{SQT} + \eta_i \geq d_i \quad \forall i \in [r]$$

$$\pi_{SQT} \geq 0 \quad \forall (S,Q,T) \in \mathfrak{T}$$

$$\mu_i, \eta_i \geq 0 \quad \forall i \in [r]$$

$$\gamma \geq 0$$

Proof: $|C^+ \cap D^+|$ odd

- WLOG, assume $f > 0$.
- Define

$$
\begin{aligned}
C^+ &:= \{k : c_k \geq 0\} \\
C^- &:= \{k : c_k < 0\} \\
D^+ &:= \{k : d_k \geq 0\} \\
D^- &:= \{k : d_k < 0\}
\end{aligned}
$$

- $\hat{u}_{C^+} = 1, \hat{u}_{C^-} = 0, \hat{v}_{D^+} = 1, \hat{v}_{D^-} = 0, \hat{z} = 1$ is optimal solution to (P) with value $c(C^+) + d(D^+) + f$.
- $\hat{\pi} = 0, \gamma = f, \hat{\mu}_{C^+} = c_{C^+}, \hat{\mu}_{C^-} = 0, \hat{\eta}_{D^+} = d_{D^+}, \hat{\eta}_{D^-} = 0$ is feasible solution to (D) with value $c(C^+) + d(D^+) + f$
- That Was Easy!

Proof: $|C^+ \cap D^+|$ Even

- Either $\hat{z} = 1$, wherein
  - Either u_k or v_k in $C^+ \cap D^+$, or
  - u_k in $C^- \cap D^+$, or
  - v_k in $C^+ \cap D^-$, or
  - Both u_k and v_k in $C^- \cap D^-$

  flip their "obvious" value to lose Δ while gaining $f > \Delta$ in the objective
- Or $\hat{z} = 0$, in which case $f < \Delta$ for all these potential elements to flip.
- Constructing a dual feasible solution (requiring $\pi_{SQT} > 0$) for all these cases (when $\hat{z} = 1$) is a tricky, four-page exercise left to the reader.

MIP Formulation 3—Monkey In Silk

$$\min \sum_{(ij)\in\Omega} |X_{ij} - z_{ij}|$$

$$(u^i, v^j, z_{ij}) \in \mathcal{I}_{ij} \quad \forall (ij) \in \Omega$$

Computational Results

MIP	% Missing	Rank	Time	Nodes	Opt
1	0	2	41.81	70237	24
2	0	2	9.42	13746	24
3	0	2	5.00	12588	24
1	0	3	7184.56	10437394	12
2	0	3	2137.15	1272534	12
3	0	3	1765.4	1962326	12
1	10	2	14.04	27757	17
2	10	2	6.63	20296	17
3	10	2	3.65	22560	17
1	10	3	320.59	996422	7
2	10	3	357.02	353021	7
3	10	3	188.81	332773	7
1	20	2	2.91	5872	14
2	20	2	3.64	8927	14
3	20	2	4.28	3357	14
1	20	3	4106.07	13393830	8
2	20	3	2199.89	2366186	8
3	20	3	381.94	645413	8

Discussion

- Frankly, the computational results are not where we want them to be.
- We can now only "reliably" solve linear index coding problems of sizes up to around 12 by 12.
- And worse, the "monkey in silk" formulation or the "pig in lipstick formulation" aren't typically much better than the "pig" formulation

---

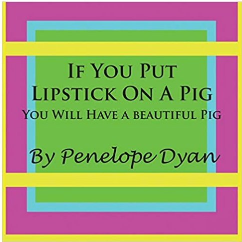
A Word on Separation

- We don't do it—Our computational results (to this point) just explicitly enumerate all inequalities
- However, separation of the SQT inequalities is "trivial" (linear time/greedy)

Can we do more?

- MIP3 (Silk Monkey) formulation is

$$(u^i, v^j, z_{ij}) \in \text{conv}(\mathcal{I}_{ij}) \quad \forall (ij) \in \Omega$$
$$(u^i, v^j, z_{ij}) \in \{0, 1\}^{dr+rn+|\Omega|}$$



- We know the intersection of the convex hulls
- If it were only true that

$$\text{conv}\left(\cap_{ij\in\Omega} \mathcal{I}_{ij}\right) = \cap_{ij\in\Omega} \text{conv}(\mathcal{I}_{ij})$$

we wouldn't need integer variables.

Next Steps: Two Rows of U

$$\mathcal{T} = \{(u, w, v, z_u, z_w) \in \{0, 1\}^{3r+2} \mid z_u = \oplus_{k=1}^{r} u_k v_k, z_w = \oplus_{k=1}^{r} w_k v_k\}$$

LOTS of Inequalities: Monkey+Pig

Monkey + Pig Inequalties: Basic Idea

- Pick two indices $\{i, j\} \in [r]$ and make two tri-partitions of $[r] \setminus \{i, j\}, (S^u, Q^u, T)$ and (S^w, Q^w, T), with $|S^u|, |S^w|$ even.
- Fix variables

$$
\begin{aligned}
u_i = v_i = 1 \ \forall i \in S^u \\
u_i = 0 \ \forall i \in Q^u \\
v_i = 0 \ \forall i \in T \\
w_i = v_i = 1 \ \forall i \in S^w \\
w_i = 0 \ \forall i \in Q^w
\end{aligned}
$$

to give the face

$$
\begin{aligned}
z_u = u_i v_i \oplus u_j v_j \\
z_w = w_i v_i \oplus w_j v_j
\end{aligned}
$$

Interesting Families

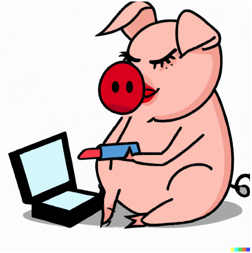
- Accounting for the symmetries where we swap $(i \leftrightarrow j)$ or $(u \leftrightarrow w)$ there are four fundamental "interesting" families of inequalities, involving both z_u and z_w

$$
\begin{aligned}
(1-u_i) + v_j + z_u + (1-z_w) &\geq 1 \\
(1-u_i) + (1-u_j) + (1-w_i) + (1-w_j) + (1-z_u) + z_w &\geq 1 \\
(1-u_i) + (1-u_j) + v_i + v_j + z_u + 2(1-z_w) &\geq 2 \\
(1-u_i) + (1-u_j) + 2(1-w_i) + (1-v_i) + v_j + (1-z_i) + 2z_j &\geq 2
\end{aligned}
$$

- Lifting each of these gives exponentially-large new families of facet-defining inequalities for the set $\mathcal{T}$.

Hard At Work—Remaining Items

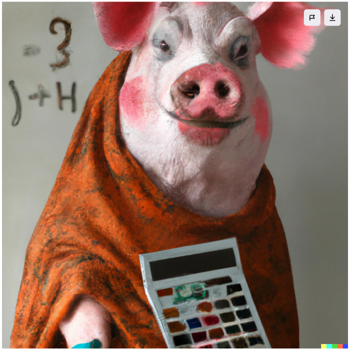
- Work out separation and implement new inequalities for $\mathcal{T}$
- Improved computational results?
- Write it up.

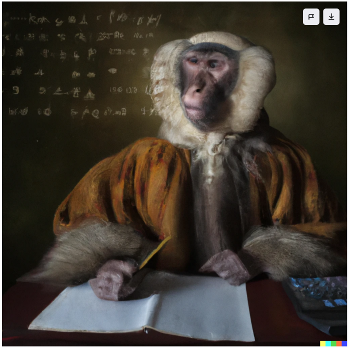


- Akhilesh has < 3 months to get it all done!

Any Questions?

- Three MIP Formulation for Matrix Completion in $\mathbb{F}_2$
- Ideas for convexifying inner products in $\mathbb{F}_2$





Another ChatGPT Apology Limerick

Write a limerick apologizing for a bad math talk

There once was a talk full of math,
But the speaker was quite in a bath,
The graphs were all skewed,
The theorems were crude,
I'm sorry for causing your wrath!