FSMV Project Presentation

Agenda

- **Exact Covering (Problem-1)**
  - Problem Statement
  - Encoding Logic
  - Output
- **D5 (Problem – 2)**
  - Problem Statement
  - Height Reduction Strategy in a 4-Bit Dadda Multiplier
  - Encoding Logic for 2 – Bit Numbers
  - Output

Exact Covering

Problem Statement

The Langford pairs problem for a given integer n asks us to place two instances of each number from 1 to n into $2n$ slots, such that for each number k, the two occurrences of k are separated by exactly k slots.

Ex. Let's take example with $n = 3$

Problem Statement

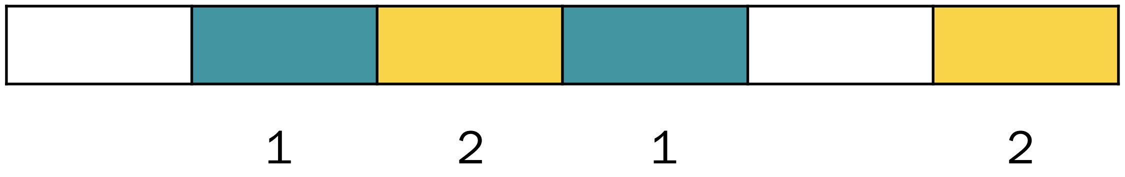
The Langford pairs problem for a given integer n asks us to place two instances of each number from 1 to n into 2n slots, such that for each number k, the two occurrences of k are separated by exactly k slots.

Ex. 3 1 2 1 3 2

Problem Statement

The Langford pairs problem for a given integer n asks us to place two instances of each number from 1 to n into 2n slots, such that for each number k, the two occurrences of k are separated by exactly k slots.

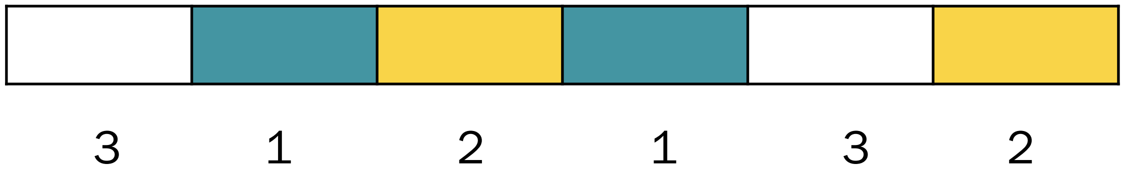
Ex. 3 1 2 1 3 2



Problem Statement

The Langford pairs problem for a given integer n asks us to place two instances of each number from 1 to n into 2n slots, such that for each number k, the two occurrences of k are separated by exactly k slots.

Ex. 3 1 2 1 3 2



Possible Positions for 1st digit

S1	S2	S3	S4	S5	S6

S1	S2	S3	S4	S5	S6

S1	S2	S3	S4	S5	S6

S1	S2	S3	S4	S5	S6

Possible Positions for 2nd digit

S1	S2	S3	S4	S5	S6
S1	S2	S3	S4	S5	S6
S1	S2	S3	S4	S5	S6

Possible Positions for 3rd digit

S1	S2	S3	S4	S5	S6

S1	S2	S3	S4	S5	S6

All Possible Positions of numbers

	D1	D2	D3	S1	S2	S3	S4	S5	S6
X1	1	0	0	1	0	1	0	0	0
X2	1	0	0	0	1	0	1	0	0
X3	1	0	0	0	0	1	0	1	0
X4	1	0	0	0	0	0	1	0	1
X5	0	1	0	1	0	0	1	0	0
X6	0	1	0	0	1	0	0	1	0
X7	0	1	0	0	0	1	0	0	1
X8	0	0	1	1	0	0	0	1	0
X9	0	0	1	0	1	0	0	0	1

How we can ignore 9th row

S1	S2	S3	S4	S5	S6

S1	S2	S3	S4	S5	S6

Clauses for the 1^{st}, 2^{nd} and 3^{rd} number

(X1 v X2 v X3 v X4)
(~X1 v ~X2) (~X1 v ~X3) (~X1 v ~X4)
(~X2 v ~X3) (~X2 v ~X4)
(~X3 v ~X4)

(X5 v X6 v X7)
(~X5 v ~X6) (~X5 v ~X7) (~X6 v ~X7)

(X8)

Clauses for S1 to S6 space

(X1 v X5 v X8)

(~X1 v ~X5) (~X1 v ~X8) (~X5 v ~X8)

(X2 v X6)

(~X2 v ~X6)

(X1 v X3 v X7)

(~X1 v ~X3) (~X1 v ~X7) (~X3 v ~X7)

(X2 v X4 v X5)

(~X2 v ~X4) (~X2 v ~X5) (~X4 v ~X5)

(X3 v X6 v X8)

(~X3 v ~X6) (~X3 v ~X8) (~X6 v ~X8)

(X4 v X7)

(~X4 v ~X7)

Encoding

```
p cnf 8 32
1 2 3 4 0
-1 -2 0
-1 -3 0
-1 -4 0
-2 -3 0
-2 -4 0
-3 -4 0
5 6 7 0
-5 -6 0
-5 -7 0
-6 -7 0
```

```
8 0
1 5 8 0
-1 -5 0
-1 -8 0
-5 -8 0
2 6 0
-2 -6 0
1 3 7 0
-1 -3 0
-1 -7 0
-3 -7 0
```

```
2 4 5 0
-2 -4 0
-2 -5 0
-4 -5 0
3 6 8 0
-3 -6 0
-3 -8 0
-6 -8 0
4 7 0
-4 -7 0
```

Output

$X2 = 1$

$X7 = 1$

$X8 = 1$

```
sugar@DESKTOP-AO833VG: ~
c
c ---- [ parsing input ] ---------------------------------------------------------
c
c reading DIMACS file from 'problem_1.cnf'
c opening file to read 'problem_1.cnf'
c found 'p cnf 8 32' header
c parsed 32 clauses in 0.00 seconds process time
c
c ---- [ options ] ---------------------------------------------------------------
c
c all options are set to their default value
c
c ---- [ solving ] ---------------------------------------------------------------
c
c time measured in process time since initialization
c
c  seconds  reductions  redundant irreductant
c         MB     restarts       trail    variables
c          level    conflicts       glue     remaining
c
c *  0.00  2  0 0   0    0    0  0% 0   19   0  0%
c l  0.00  2  0 0   0    0    0  0% 0   19   0  0%
c 1  0.00  2  0 0   0    0    0  0% 0   19   0  0%
c
c ---- [ result ] ----------------------------------------------------------------
c
s SATISFIABLE
v -1 2 -3 -4 -5 -6 7 8 0
c
c ---- [ run-time profiling ] ----------------------------------------------------
c
c process time taken by individual solving procedures
c (percentage relative to process time for solving)
c
c        0.00   41.69% parse
c        0.00   15.56% search
c        0.00   15.13% lucky
```

Problem Statement D5

We aim to prove $(X1...X9)_2 \times (Y1...Y9)_2 \neq (X1\ ...X9)_2 \times' (Y1\ ...Y9)_2$, with two copies of the same Dadda multiplication circuit which is an unsatisfiability problem.

Dadda Multiplication Circuit

Dadda circuits are optimized for fast binary multiplication, minimizing the number of addition operations required.

Height Reduction Strategy in a 4-Bit Dadda Multiplier

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
			A_3B_0	A_2B_0	A_1B_0	A_0B_0
		A_3B_1	A_2B_1	A_1B_1	A_0B_1	
	A_3B_2	A_2B_2	A_1B_2	A_0B_2		
A_3B_3	A_2B_3	A_1B_3	A_0B_3			

Height Reduction Strategy in a 4-Bit Dadda Multiplier

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
A_3B_3	A_3B_2	A_3B_1	A_3B_0	A_2B_0	A_1B_0	A_0B_0
	A_2B_3	A_2B_2	A_2B_1	A_1B_1	A_0B_1	
		A_1B_3	A_1B_2	A_0B_2		
			A_0B_3			

Height Reduction Strategy in a 4-Bit Dadda Multiplier

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
A_3B_3	A_3B_2	A_3B_1	A_3B_0	A_2B_0	A_1B_0	A_0B_0
	A_2B_3	A_2B_2	A_2B_1	A_1B_1	A_0B_1	
		A_1B_3	A_1B_2	A_0B_2		
			A_0B_3			

Height Reduction Strategy in a 4-Bit Dadda Multiplier

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
A_3B_3	C_2	C_1	S_1	A_2B_0	A_1B_0	A_0B_0
	A_3B_2	S_2	A_1B_2	A_1B_1	A_0B_1	
	A_2B_3	A_1B_3	A_0B_3	A_0B_2		

Height Reduction Strategy in a 4-Bit Dadda Multiplier

			A_3	A_2	A_1	A_0
			B_3	B_2	B_1	B_0
C_6	C_5	C_4	C_3	S_3	A_1B_0	A_0B_0
A_3B_3	S_6	S_5	S_4	A_0B_2	A_0B_1	

Height Reduction Strategy in a 4-Bit Dadda Multiplier

				A_3	A_2	A_1	A_0
				B_3	B_2	B_1	B_0
	C_{11}	C_{10}	C_9	C_8	C_7	A_1B_0	A_0B_0
	C_6	C_5	C_4	C_3	S_3	A_0B_1	
	A_3B_3	S_6	S_5	S_4	A_0B_2		
C_{12}	S_{12}	S_{11}	S_{10}	S_9	S_8	S_7	A_0B_0

Dadda Multiplication for 2 – bit Number

$$
\begin{array}{cccc}
 & & A_1 & A_0 \\
 & & B_1 & B_0 \\
\hline
 & & (B_0 \wedge A_1) & (B_0 \wedge A_0) \\
 & (B_1 \wedge A_1) & (B_1 \wedge A_0) & \\
\hline
S_3 & S_2 & S_1 & S_0
\end{array}
$$

$S_0 = (B_0 \wedge A_0)$

$S_1 = (B_0 \wedge A_1) \oplus (B_1 \wedge A_0)$

$C_1 = (B_0 \wedge A_1) \wedge (B_1 \wedge A_0)$

$S_2 = C_1 \oplus (B_1 \wedge A_1)$

$C_2 = C_1 \wedge (B_1 \wedge A_1)$

$S_3 = C_2$

Dadda Multiplication for 2 – bit Number

1. **Encoding** $S_0 = B_0 \wedge A_0$

For $S_0 = B_0 \wedge A_0$:

- $S_0 \to (B_0 \wedge A_0)$: $(\neg S_0 \vee B_0) \wedge (\neg S_0 \vee A_0)$
- $(B_0 \wedge A_0) \to S_0$: $\neg B_0 \vee \neg A_0 \vee S_0$

So, the CNF clauses are:

$$(\neg S_0 \vee B_0) \wedge (\neg S_0 \vee A_0) \wedge (\neg B_0 \vee \neg A_0 \vee S_0)$$

2. Encoding $S_1 = (B_0 \wedge A_1) \oplus (B_1 \wedge A_0)$

Introduce an intermediate variable $X_1 = B_0 \wedge A_1$:

- $X_1 \rightarrow (B_0 \wedge A_1)$: $(\neg X_1 \vee B_0) \wedge (\neg X_1 \vee A_1)$
- $(B_0 \wedge A_1) \rightarrow X_1$: $\neg B_0 \vee \neg A_1 \vee X_1$

Introduce another intermediate variable $X_2 = B_1 \wedge A_0$:

- $X_2 \rightarrow (B_1 \wedge A_0)$: $(\neg X_2 \vee B_1) \wedge (\neg X_2 \vee A_0)$
- $(B_1 \wedge A_0) \rightarrow X_2$: $\neg B_1 \vee \neg A_0 \vee X_2$

Now, $S_1 = X_1 \oplus X_2$:

- $S_1 \rightarrow (X_1 \vee X_2) \wedge (\neg X_1 \vee \neg X_2)$: $(\neg S_1 \vee X_1 \vee X_2) \wedge (\neg S_1 \vee \neg X_1 \vee \neg X_2)$
- $(X_1 \vee X_2) \wedge (\neg X_1 \vee \neg X_2) \rightarrow S_1$: $(X_1 \vee \neg X_2 \vee S_1) \wedge (\neg X_1 \vee X_2 \vee S_1)$

3. Encoding $C_1 = (B_0 \wedge A_1) \wedge (B_1 \wedge A_0)$

Using X_1 and X_2 from above:

- $C_1 \rightarrow (X_1 \wedge X_2)$: $(\neg C_1 \vee X_1) \wedge (\neg C_1 \vee X_2)$
- $(X_1 \wedge X_2) \rightarrow C_1$: $(\neg X_1 \vee \neg X_2 \vee C_1)$

4. Encoding $S_2 = C_1 \oplus (B_1 \wedge A_1)$

Introduce $X_3 = B_1 \wedge A_1$:

- $X_3 \to (B_1 \wedge A_1)$: $(\neg X_3 \vee B_1) \wedge (\neg X_3 \vee A_1)$
- $(B_1 \wedge A_1) \to X_3$: $(\neg B_1 \vee \neg A_1 \vee X_3)$

Now, $S_2 = C_1 \oplus X_3$:

- $S_2 \to (C_1 \vee X_3) \wedge (\neg C_1 \vee \neg X_3)$: $(\neg S_2 \vee C_1 \vee X_3) \wedge (\neg S_2 \vee \neg C_1 \vee \neg X_3)$
- $(C_1 \vee X_3) \wedge (\neg C_1 \vee \neg X_3) \to S_2$: $(C_1 \vee \neg X_3 \vee S_2) \wedge (\neg C_1 \vee X_3 \vee S_2)$

5. Encoding $C_2 = C_1 \wedge (B_1 \wedge A_1)$

Using X_3 from above:

- $C_2 \rightarrow (C_1 \wedge X_3)$: $(\neg C_2 \vee C_1) \wedge (\neg C_2 \vee X_3)$
- $(C_1 \wedge X_3) \rightarrow C_2$: $(\neg C_1 \vee \neg X_3 \vee C_2)$

6. Encoding $S_3 = C_2$

For $S_3 = C_2$:

- $S_3 \rightarrow C_2$: $(\neg S_3 \vee C_2)$
- $C_2 \rightarrow S_3$: $(\neg C_2 \vee S_3)$

Final clauses (28)

$$(\neg S_0 \vee B_0) \wedge (\neg S_0 \vee A_0) \wedge (\neg B_0 \vee \neg A_0 \vee S_0) \wedge$$

$$(\neg X_1 \vee B_0) \wedge (\neg X_1 \vee A_1) \wedge (\neg B_0 \vee \neg A_1 \vee X_1) \wedge (\neg X_2 \vee B_1) \wedge (\neg X_2 \vee A_0) \wedge (\neg B_1 \vee \neg A_0 \vee X_2) \wedge$$

$$(\neg S_1 \vee X_1 \vee X_2) \wedge (\neg S_1 \vee \neg X_1 \vee \neg X_2) \wedge (X_1 \vee \neg X_2 \vee S_1) \wedge (\neg X_1 \vee X_2 \vee S_1) \wedge$$

$$(\neg C_1 \vee X_1) \wedge (\neg C_1 \vee X_2) \wedge (\neg X_1 \vee \neg X_2 \vee C_1) \wedge$$

$$(\neg X_3 \vee B_1) \wedge (\neg X_3 \vee A_1) \wedge (\neg B_1 \vee \neg A_1 \vee X_3) \wedge$$

$$(\neg S_2 \vee C_1 \vee X_3) \wedge (\neg S_2 \vee \neg C_1 \vee \neg X_3) \wedge (C_1 \vee \neg X_3 \vee S_2) \wedge (\neg C_1 \vee X_3 \vee S_2) \wedge$$

$$(\neg C_2 \vee C_1) \wedge (\neg C_2 \vee X_3) \wedge (\neg C_1 \vee \neg X_3 \vee C_2) \wedge$$

$$(\neg S_3 \vee C_2) \wedge (\neg C_2 \vee S_3)$$

Output of SAT Solver

```
c
c ---- [ parsing input ] -----------------------------------------------------
c
c reading DIMACS file from 'problem_D5.cnf'
c opening file to read 'problem_D5.cnf'
c found 'p cnf 511 2881' header
c parsed 2881 clauses in 0.00 seconds process time
c
c ---- [ options ] -----------------------------------------------------------
c
c all options are set to their default value
c
c ---- [ solving ] -----------------------------------------------------------
c
c time measured in process time since initialization
c
c  seconds  reductions  redundant irreducible
c         MB    restarts        trail    variables
c          level   conflicts       glue      remaining
c
c *  0.00  4  0 0   0    0    0  0% 0 2881 511 100%
c {  0.01  4  0 0   0    0    0  0% 0 2881 511 100%
c i  0.01  4  1 0   0    1    0 14% 0 2881 510 100%
c i  0.01  4  1 0   0    2    0 15% 0 2881 479 94%
c i  0.01  4  1 0   0    3    0 16% 0 2881 478 94%
c i  0.01  4  1 0   1    4    0 17% 0 2881 477 93%
c i  0.01  4  1 0   1    5    0 17% 0 2881 476 93%
c i  0.01  4  1 0   1    6    0 17% 0 2881 475 93%
c i  0.01  4  1 0   2    7    0 17% 0 2881 474 93%
c i  0.01  4  1 0   2    8    0 17% 0 2881 473 93%
c i  0.01  4  1 0   2    9    0 16% 0 2881 472 92%
c i  0.01  4  1 0   3   10    0 16% 0 2881 469 92%
c i  0.01  4  1 0   3   11    0 16% 0 2881 467 91%
c i  0.01  4  1 0   3   12    0 16% 0 2881 465 91%
c i  0.01  4  1 0   4   13    0 16% 0 2881 463 91%
c }  0.01  4  1 0   4   16    0 16% 0 2881 408 80%
```

```
sugar@DESKTOP-AO833VG: ~
c ---- [ result ] ------------------------------------------------------------
c
s UNSATISFIABLE
c
c ---- [ run-time profiling ] -------------------------------------------------
c
c process time taken by individual solving procedures
c (percentage relative to process time for solving)
c
c         0.00  105.46% parse
c         0.00   92.72% search
c         0.00   75.68% unstable
c         0.00   12.74% lucky
c         0.00    0.00% simplify
c   =================================
c         0.00   46.19% solve
c
c last line shows process time for solving
c (percentage relative to total process time)
c
c ---- [ statistics ] --------------------------------------------------------
c
c chronological:                  14        87.50 %  of conflicts
c conflicts:                      16      3974.17    per second
c decisions:                     215     53402.88    per second
c fixed:                         103        20.16 %  of all variables
c learned:                        15        93.75 %  per conflict
c learned_lits:                   15       100.00 %  learned literals
c minimized:                       0         0.00 %  learned literals
c shrunken:                        0         0.00 %  learned literals
c minishrunken:                    0         0.00 %  learned literals
c otfs:                            0         0.00 %  of conflict
c propagations:                  686         0.17 M  per second
c restarts:                        4         4.00    interval
c
c seconds are measured in process time for solving
c
```

Thank You