Langford's Problem

Moti Ben-Ari

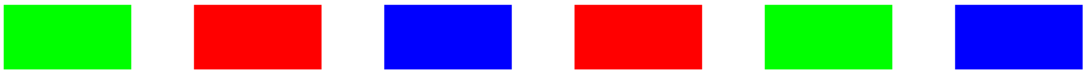
Department of Science Teaching

Weizmann Institute of Science

http://www.weizmann.ac.il/sci-tea/benari/

The Definition of Langford's problem

Scottish mathematician C. Dudley Langford noticed that his son had arranged colored blocks in the following arrangement:



There is one block between the two red blocks, two blocks between the blue blocks and three blocks between the green blocks. Therefore, the problem can be expressed as follows:

> Given the bag of numbers {1,1,2,2,3,3}, can they be arranged in a sequence such that for $1 \leq i \leq 3$ there are i numbers between the two occurrences of i?[1]

From the arrangement of the colored blocks, we see that a solution is 312132.

The generalized problem is:

> **Langford's Problem L(n)** Given the bag of numbers {1,1,2,2,3,3,...,n,n}, can they be arranged in a sequence such that for $1 \leq i \leq n$ there are i numbers between the two occurrences of i?

Langford's problem as a covering problem

Langford's problem can be posed using arrays. For $L(3)$, there are 6 columns, one for each of the $2 \cdot 3$ numbers. The rows are defined by the definition of the problem: the two occurrences of k must have k numbers between them. It is easy to see that there are four possible placements of 1, three of 2 and two of 3:

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

To solve the problem, we need to select *one* row for the 1's in the sequence, *one* row for the 2's and *one* row for the 3's, such that if we stack these rows on top of each other, no column contains more than one number:

[1]A *bag* is like a set except that duplicate elements may appear.

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

First, note that row 9 is not needed because of symmetry: starting with row 9 just gives the reversal of the sequence obtained by starting with row 8.

Row 8 is the only one containing 3's so it must be chosen and the result is 3␣␣␣3␣. Any row with numbers in columns 1 and 5 can no longer be used, because only one number can be placed at each position. Let us denote the permissible and forbidden rows by $\cancel{1}, 2, \cancel{3}, 4, \cancel{5}, \cancel{6}, 7, 8$.

Row 7 is the only remaining row containing 2's so must be chosen and the result is 3␣2␣32. Deleting rows that can no longer be used gives: $\cancel{1}, 2, \cancel{3}, \cancel{4}, \cancel{5}, \cancel{6}, 7, 8$.

Choosing the only remaining row, row 2, gives the solution 312132. Furthermore, the analysis has shown that this is the only solution.

Langford's problem L(4)

Here is the array for $L(4)$:

	1	2	3	4	5	6	7	8
1	1		1					
2		1		1				
3			1		1			
4				1		1		
5					1		1	
6						1		1
7	2			2				
8		2			2			
9			2			2		
10				2			2	
11					2			2
12	3				3			
13		3				3		
14			3				3	
15				3				3
16	4					4		
17		4					4	
18			4					4

We leave it to the reader to show that the only solution is 41312432.

For which values of n is Langford's problem solvable?

Theorem $L(n)$ has a solution if and only if $n = 4k$ or $n = 4k - 1$.

We give two proofs of the forward implication based on Miller (2014).. For the converse, see Davies (1959).

Proof 1 If the first occurrence of the number k is at position i_k, the second occurrence is at position $i_k + k + 1$. The sum of the positions of all the numbers is:

$$S_n = \sum_{k=1}^{n} i_k + \sum_{k=1}^{n} (i_k + k + 1) .$$

But S_n, the sum of the positions, is simply $1 + 2 + 3 + \cdots + 2n$, so:

$$S_n = \sum_{k=1}^{2n} k = \frac{2n(2n+1)}{2} ,$$

using the formula for the sum of an arithmetic progression. Simplifying:

$$S_n = \sum_{k=1}^{n} i_k + \sum_{k=1}^{n} (i_k + k + 1) = 2 \sum_{k=1}^{n} i_k + \sum_{k=1}^{n} (k+1) = 2 \sum_{k=1}^{n} i_k + \frac{n(n+3)}{2} .$$

Equating the two formulas for S_n gives:

$$2 \sum_{k=1}^{n} i_k + \frac{n(n+3)}{2} = \frac{2n(2n+1)}{2} ,$$

and:

$$\sum_{k=1}^{n} i_k = \frac{1}{2} \left(\frac{2n(2n+1)}{2} - \frac{n(n+3)}{2} \right) = \frac{3n^2 - n}{4} .$$

The left-hand side is the sum of integers (the positions), so it must be an integer. The right-hand side must also be an integer. When is $3n^3 - n$ divisible by 4? Factoring gives $3n^2 - n = n(3n-1)$, so if n is a multiple of 4, the product is divisible by 4.

When is $3n - 1$ divisible by 4? Any integer n can be expressed as $n = 4i + j$ for $j = 0, 1, 2, 3$. If $3n - 1$ is divisible by 4, then so is $3(4i + j) - 1 = 12i + 3j - 1$. Clearly, $12i$ is divisible by 4, and it is easy to see that $3j - 1$ is divisible by 4 (for $j = 0, 1, 2, 3$) only if $j = 3$, that is, $n = 4i + 3 = 4(i + 1) - 1$.

Proof 2 Consider the solution for $n = 4$:

$$\begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 4 & 1 & 3 & 1 & 2 & 4 & 3 & 2 \end{array}$$

The positions of the occurrences of 4 are 1 and 6, and the positions of the occurrences of 2 are 5 and 8. One position is odd and one is even. In the general case, if i is the position of the first occurrence of an *even* number k, then the position of the second occurrence is $i + k + 1$. Since k is even, $k + 1$ is odd. Since odd plus odd is even and even plus odd is odd, one of i, $i + k + 1$ must be odd and the other even.

The positions of the occurrences of 1 are 2 and 4 and the positions of the occurrences of 3 are 3 and 7. For any *odd* number k, $k + 1$ is even, so if i is even, then $i + k + 1$ is even, and if i is odd, then $i + k + 1$ is odd.

Obviously, the positions $1, 2, \ldots, 2n - 1, 2n$ contain an equal number of even and odd positions. When placing the two occurrences of a number in the sequence, they "take over" two positions. When the sequence is complete, there must be an equal number of even and odd positions "taken over." We use the term *parity* for the difference between the number of even and odd positions taken. Initially, the parity is zero, and if the problem has a solution, the completed sequence also has a parity of zero.

When the two occurrences of an even number is placed, they take over one even position and one odd position, so the parity remains the same. When the two occurrences of an odd number is placed, the parity becomes $+2$ or -2, so we must be able to associate this pair with *another* odd pair placed at positions that balance out the parity. In other words, there must be an even number of pairs of odd numbers, that is, *there must be an even number of odd numbers in* $\{1, \ldots, n\}$.

The theorem claims that if there is a solution, either $n = 4k$ or $n = 4k - 1$, and if there is no solution, then either $n = 4k - 2$ or $4k - 3$. The proof is by induction. For the base case, $k = 1$, it is easy to see that the sets $\{1\}$ and $\{1, 2\}$ have no solution; furthermore, they have an odd number of odd numbers. For the sets $\{1, 2, 3\}$ and $\{1, 2, 3, 4\}$, we showed that they have solutions; furthermore, they have an even number of odd numbers.

The inductive hypothesis is that solutions are possible for $\{1, \ldots, 4k - j\}$, $k \geq 1$, $j = 0, 1$ (and they have an even number of odd numbers), and solutions are impossible for $j = 2, 3$ (and they have an odd number of odd numbers). Since $4k + 1$ is odd, adding it increases the number of odd numbers by one, so there is no solution. Similarly, adding $4k + 1, 4k + 2$ has no solution since $4k + 1$ is odd and $4k + 2$ is even. Adding $4k + 1, 4k + 2, 4k + 3$ or $4k + 1, 4k + 2, 4k + 3, 4k + 4$ adds two odd numbers so there is still an even number of odd numbers and there can be a solution. We have proved the claim for $4(k + 1) - j$ and by induction the theorem holds for all $n \geq 1$.

SAT solving

In propositional logic, an assignment of `true` and `false` to the atomic propositions of a formula A *satisfies* A if A evaluates to `true`. A SAT solver is a computer program that checks if a formula in CNF is satisfiable or unsatisfiable. (For an overview of SAT solving, see Ben-Ari (2012), Chapter 6.) Knuth (2015) shows how solutions to Langford's problem can be found by a SAT solver, by encoding the array representation as a CNF formula.

Let x_i be true if row i is chosen. For $L(3)$, the following clauses encode that exactly one of rows 1–4 containing 1 must be chosen:

```
[x1,x2,x3,x4],
[~x1,~x2], [~x1,~x3], [~x1,~x4], [~x2,~x3], [~x2,~x4], [~x3,~x4]
```

The first clause encodes that at least one row must be chosen. The next clause encodes that if row 1 is chosen, $x_1 =$ `true`, then row 2 cannot be chosen, $x_2 =$ `false`, and similarly for the other pairs of rows. Other clauses encode that exactly one of rows 5, 6, 7 must be chosen and that the row 8 must be chosen.

Clauses are also needed to express the constraints on the columns. For example, column 1 requires that exactly one of rows 1, 5, 8 be chosen:

```
[x1,x5,x8], [~x1,~x5], [~x1,~x8], [~x5,~x8]
```

Running a SAT solver returns the solution that rows 2, 7 and 8 should be chosen:

```
Satisfying assignments: [x1=0,x2=1,x3=0,x4=0,x5=0,x6=0,x7=1,x8=1]
```

CNF formulas for $L(3)$ and $L(4)$ can be found in the archive for LearnSAT, a SAT solver I developed for teaching. On my website `http://www.weizmann.ac.il/sci-tea/benari/` follow the links for "Software and Learning Materials" and then "LearnSAT".

References

M. Ben-Ari. *Mathematical Logic for Computer Science (Third Edition)*, Springer, 2012.

R.O. Davies. On Langford's problem (II), *Mathematical Gazette*, 43, 1959, 253-5.

D.E. Knuth. *The Art of Computer Programming, Volume 4, Fascicle 6: Satisfiability*, Pearson, 2015.

J.E. Miller. *Langford's Problem, Remixed*, `http://dialectrix.com/langford.html`, 2014.