Variations on the Missionaries and Cannibals Puzzle

George SPAHN and Doron ZEILBERGER

Abstract: We explore both automated and human approaches to the generalized Missionaries and Cannibals problem.

A Classical Puzzle

Many of us have heard, and quite a few of us, completely solved, the following famous puzzle ([Wi], and references thereof).

three missionaries and three cannibals must cross a river using a boat which can carry at most two people, under the constraint that, for both banks, if there are missionaries present on the bank, they cannot be outnumbered by cannibals (if they were, the cannibals would eat the missionaries). The boat cannot cross the river by itself with no people on board.

This problem was also used a challenge at the very early stages of AI (and also used in Engineering Design, see the wonderful book [D] (pp. 139-143) by the brother of the advisor of one of us)

As indicated in [Wi] we set up a directed graph with the vertices labeled by triples [m,c,b] where

- m is the number of missionaries currently at the first bank
- c is the number of cannibals currently at the first bank
- $b = 1$ if the boat is currently at the first bank, and $b = 0$ if it is at the second bank.

It follows that the numbers of missionaries and cannibals in the second bank are $3 - m$ and $3 - c$ respectively.

In order for no missionary to be eaten, we need at all times:

- If $m > 0$ then $m \geq c$
- If $3 - m > 0$ then $3 - m \geq 3 - c$, or equivalently if $m < 3$ then $c \geq m$.

The edges are

- $[m, c, 1] \to [m - e_1, c - e_2, 0]$
- $[m, c, 0] \to [m + e_1, c + e_2, 1]$

where $0 < e_1 + e_2 \leq 2$ and both vertices are legal, where the boat carries e_1 missionaries and e_2 cannibals.

To solve the puzzle all we need is find a shortest path from the initial state $[3, 3, 0]$ to the final state

$[0, 0, 1]$.

Part I: Let the Computer Do it

Now that we have such high-level languages (Maple in our case), it is an easy programming exercise to solve not only the original puzzle, but the following general version, for any **specific, numerical parameters** M, C, B, and d, and not only find **one** solution but **all of them**.

So we set up to write Maple code for finding **all** solutions for any inputted M, C, B, d, the following kind of puzzles.

M missionaries and C cannibals must cross a river using a boat which can carry at most B people, under the constraint that, for both banks, if there are missionaries present in a bank, the number of missionaries must exceed the number of cannibals but at least d. (if $d > 0$ this means that the cannibals are stronger than the missionaries, so one needs a higher 'safety margin')

Now the number of missionaries and cannibals in the second bank are $M - m$ and $C - c$ respectively.

Now, in order for no missionary to be eaten we need at all times

- If $m > 0$ then $m - c \geq d$
- If $M - m > 0$ then $(M - m) - (C - c) \geq d$

The edges are:

$[m, c, 1] \rightarrow [m - e_1, c - e_2, 0]$

$[m, c, 0] \rightarrow [m + e_1, c + e_2, 1]$

where $0 < e_1 + e_2 \leq B$ and both vertices are legal, where the boat carries e_1 missionaries and e_2 cannibals.

This is implemented in procedure

`Sols(M,C,B,d)`

in our Maple package `Cannibals,txt` obtainable free of charge from the front of this article

`https://sites.math.rutgers.edu/~zeilberg/mamarim/mamarimhtml/cannibals.html` .

In particular to get all four solutions for the original puzzle, type

`Sols(3,3,2,0);`

and get in 0.04 seconds the four solutions

[[3, 3, 1], [2, 2, 0], [3, 2, 1], [3, 0, 0], [3, 1, 1], [1, 1, 0], [2, 2, 1],[0, 2, 0], [0, 3, 1], [0, 1, 0], [0, 2, 1], [0, 0, 0]]

[[3, 3, 1], [2, 2, 0],[3, 2, 1], [3, 0, 0], [3, 1, 1], [1, 1, 0], [2, 2, 1], [0, 2, 0], [0, 3, 1], [0, 1, 0], [1, 1, 1], [0, 0, 0]]

[[3, 3, 1], [3, 1, 0], [3, 2, 1], [3, 0, 0], [3, 1, 1], [1, 1, 0], [2, 2, 1], [0, 2, 0], [0, 3, 1], [0, 1, 0], [0, 2, 1], [0,0, 0]]

[[3, 3, 1], [3, 1, 0], [3, 2, 1], [3, 0, 0], [3, 1, 1], [1, 1, 0], [2,2, 1], [0, 2, 0], [0, 3, 1], [0, 1, 0], [1, 1, 1], [0, 0, 0]]

For any solution, `S`, procedure `SO(M,C,B,d,S)` spells it out, very verbosely. See the output file

`https://sites.math.rutgers.edu/~zeilberg/tokhniot/oCannibals1.txt` .

To see all 25 solutions with 5 missionaries and 5 cannibals, boat-capacity 3 and safety-margin 0, as well as the spelled-out solution for the first one see

`https://sites.math.rutgers.edu/~zeilberg/tokhniot/oCannibals2.txt`

The above-mentioned front of this article contains a few more sample solutions, but readers (who have access to Maple) are welcome to solve many other cases.

Part II: Let Humans do some General Thinking

We now consider the problem of determining for which values of the parameters there exists a solution. Again we have:

• M, the number of missionaries, should be at least 1

• C, the number of cannibals, should be at least 1

• B, the boat size, should be at least 2

• d, if both cannibals and missionaries are present (on either side of the river or in the boat) then the number of missionaries must be at least the number of cannibals plus d.

We will now describe some strategies for solving the puzzle.

Two Boat: This strategy works for any boat size, and never requires missionaries and cannibals to be in the boat at the same time. A single missionary can be sent across by sending 2 across and then having one come back with the boat. In this manner, we send as many missionaries across as possible, 1 at a time, and then we alternate sending a cannibal across with sending a missionary across. Without violating the rule, we can send over $M - C - d - 1$ missionaries across at the start, and bring the boat back. Then we can no longer send over 2 missionaries, so we send over two cannibals with the intent of having 1 come back with the boat. This will not violate the rule as long as $(M - C - d - 1) \geq 2 + d$ which simplifies to:

$$M - C \geq 2d + 3$$

From there it is easy to alternate sending over missionaries with cannibals until the puzzle is complete.

Big Boat: If the boat is big enough relative to the number of Cannibals, we can send all of the missionaries across before sending any cannibals across. Similar to Two Boat, we start by sending $M-C-d-1$ missionaries across and bringing the boat back. The number of missionaries remaining is then $C + d + 1$. If

$$B \geq C + d + 1$$

we can proceed to send the rest of the missionaries over, send $B-1$ back, use them to ferry a single cannibal across, send the cannibal back, and then the cannibals can all ferry themselves across.

Big Boat 2: If the boat is so big that it can carry all the missionaries in it, that also works. Send 2 cannibals across, send 1 back, send all the missionaries over, send 1 cannibal back, send the rest of the cannibals across.

$$B \geq M, C \geq 2$$

Split Cannibals: In this strategy we send half of the cannibals over, and then all the missionaries over. It is similar to Big Boat in that it only works if the Boat is large compared to half the number of cannibals. The strategy is slightly different depending on if there is an even number or odd number of cannibals. First consider the even case. We send over half the cannibals. Then we can send a boat of $C/2 + d + 1$ missionaries. This requires

$$B > C/2 + d + 1$$

and

$$M - (C/2 + d + 1) \geq C/2 + d$$

which simplifies to

$$M - C \geq 2d + 1$$

Then the missionaries can ferry themselves across, and then the cannibals can ferry themselves across.

In the odd case, we send $\lceil C/2 \rceil$ cannibals across, then $\lceil C/2 \rceil + d + 1$ missionaries across, and we can proceed similarly if

$$B > \lceil C/2 \rceil + d + 1$$

and

$$M - (\lceil C/2 \rceil + d + 1) \geq \lfloor C/2 \rfloor + d$$

which simplifies to

$$M - C \geq 2d + 1$$

However, it is still sometimes possible if

$$B = \lfloor C/2 \rfloor + d + 1$$

This is an open problem.

Simultaneous Ferry: Here we start by sending d missionaries over, and then we repeatedly send $d + 1$ missionaries and 1 cannibal over, and d missionaries back. Since cannibals and missionaries are in all 3 places at once, this requires

$$M - C \geq 3d$$

in addition to

$$B \geq d + 2$$

For each of the above strategies, we get a sufficient condition for when the puzzle is solvable. Summarizing them:

Two Boat	$M - C \geq 2d + 3$
Big Boat 1	$B \geq C + d + 1$
Big Boat 2	$B \geq M$ AND $C \geq 2$
Split Can	$M - C \geq 2d + 1$ AND $B > \lceil C/2 \rceil + d + 1$
Simul	$M - C \geq 3d$ AND $B \geq d + 2$

Also note that the condition that $M - C \geq d$ is assumed.

The above strategies do not say too much about what happens when $d = 0$. We tackle this case separately. We first give a simple strategy if $d = 0$ and $M > C$.

$d = 0, M > C$**:** Send over a boat with 1 missionary and 1 cannibal. Send 1 cannibal back. Send over 1 missionary and 1 cannibal. Send 1 missionary back. Repeat this strategy until all the cannibals are across, then have the missionaries ferry the remaining missionaries across.

When $M = C$ and $B \geq 4$, another simple strategy applies.

$d = 0, M = C, B \geq 4$**:** Send over 2 missionaries and 2 cannibals. Send 1 missionary and 1 cannibal back. Repeat until everyone is across.

The remaining cases are when $M = C$, and either $B = 2$ or $B = 3$. For $B = 2$ it turns out to be doable when $M \leq 3$ and for $B = 3$ it turns out to be doable when $M \leq 5$. These cases are not easily described by a simple strategy and make for a fun puzzle.

Necessary conditions? Are the above conditions necessary for the riddle to be solved? Not entirely. One example is that sometimes the Split cannibals strategy can still be applied when the condition is not satisfied. With more computation power, we should get a better idea of any potential cases where a strategy not described can be used.

References

[D] Clive L. Dym, *"Engineering Design, A Synthesis of Views"*, Cambridge University Press, 1994.

[Wi] Wikipedia, *Missionaries and cannibals problem*.

George Spahn and Doron Zeilberger, Department of Mathematics, Rutgers University (New Brunswick), Hill Center-Busch Campus, 110 Frelinghuysen Rd., Piscataway, NJ 08854-8019, USA.
Email: `gs828 at math dot rutgers dot edu` , `DoronZeil] at gmail dot com` .

First Written: (**Sept. 21, 2022**).