

A Note on the Iterative Algorithm for the Reve's Puzzle

A. A. K. MAJUMDAR

Department of Mathematics, Bangladesh University of Engineering and Technology (BUET),
Dhaka 1000, Bangladesh

An iterative algorithm based on the divide-and-conquer approach has been presented by van de Liefvoort for the Reve's puzzle. This note justifies the method suggested by Liefvoort to determine the number and sizes of the optimal slices, each of which can be moved in some iterative algorithm of the three-peg problem.

Received October 1993, revised February 1994

1. INTRODUCTION

In a recent paper, van de Liefvoort (1992) treated the four-peg Tower of Hanoi problem, commonly known as the Reve's puzzle. He gave a divide-and-conquer approach of the problem. The idea is to divide the tower of n discs into m slices of sizes s_1, s_2, \dots, s_m with

$$0 < s_1 < s_1 + s_2 < \dots < s_1 + s_2 + \dots + s_m = n,$$

where s_1 is the number of discs in the bottom slice, and s_m is the number of discs in the top slice and is such that the corresponding Reve's puzzle with s_m discs is solvable linearly [so that, by Proposition 2 of Newman-Wolfe (1986), $s_m \leq 3$] in $(2s_m - 1)$ number of moves. Starting with the bottom, the slices are numbered $1, 2, \dots, m$ and each slice is then identified by the associated slice number. Except for the topmost (smallest) slice, each of the remaining slices is then moved as a single entity in the manner of the three-peg algorithm, one such being provided by van de Liefvoort himself.

Van de Liefvoort (1992) suggested two alternative methods to find the number of slices, m , and the number of discs in each slice. In one version, adopted in his algorithm, the sizes of the slices are determined as follows: starting with the topmost slice, choose the sizes to be $1, 2, \dots, m$, until there are not enough discs leftover (r) to form the next slice of size $(m + 1)$. If $r = 0$ [so that $n = m(m + 1)/2$], then one is done, and in this case, the tower is 'saturated'; if, on the other hand, $r > 0$, then give one to each of the r largest slices (corresponding to the saturated case).

Though the algorithm of van de Liefvoort (1992) is different and new from the previous iterative algorithms, such as those of Rohl and Gedeon (1986), Lu (1989), Hinz (1989) and the recent one of Gedeon (1992), the various results of van de Liefvoort lack the support of the necessary theoretical developments.

This note derives the method suggested by van de Liefvoort to determine the number and sizes of the optimal slices.

2. REVE'S PUZZLE

Denoting by $M(n)$ the number of moves in a pms (presumed minimal solution) for the Reve's puzzle with n discs, the dynamic programming equation satisfied by $M(n)$ is (see, e.g. Hinz, 1989)

$$M(n) = \min_{1 \leq k \leq n-1} \{2M(k) + 2^{n-k} - 1\}, \quad n \geq 2, \quad (1)$$

$$M(0) = 0, \quad M(1) = 1. \quad (2)$$

Denoting by $f(n)$ a minimum partition number (for which the term inside the braces on the right-hand side of (1) is minimized), we have the following theorem, due to Hinz (1989).

THEOREM 1. Let, for some $l \in \mathbb{N} = \{0, 1, 2, \dots\}$,

$$\binom{l+1}{2} \leq n < \binom{l+2}{2} \quad (3)$$

so that

$$n = \binom{l+1}{2} + r \quad \text{for some } r \in \{0, 1, \dots, l\}. \quad (4)$$

Then

$$f(n) = \begin{cases} \binom{l}{2}, & \text{if } r = 0 \\ \binom{l}{2} + r - 1, & \text{if } r \neq 0 \end{cases}$$

It may be mentioned here that given $n \in \mathbb{N}$, there is only a unique l satisfying the inequality (3).

Now, regarding $f(n)$ as the function $f: \mathbb{N} \rightarrow \mathbb{N}$, we denote by f^p the p -fold composition, that is,

$$f^p(n) = \underbrace{f(f(\dots(f(n))))}_{p \text{ factors}}; \quad n, p \in \mathbb{N} \quad (5)$$

(with $f^0(n) = n$, $f^n(0) = 0$). We then have the following theorem.

THEOREM 2. Let, for some $l \in \mathbb{N}$,

$$n = \binom{l+1}{2} + r, \quad r \in \{0, 1, \dots, l\}.$$

Then, for $p \in \{0, 1, \dots, l\}$,

$$f^p(n) = \begin{cases} \binom{l-p+1}{2} + r - p, & \text{if } 0 \leq p \leq r \\ \binom{l-p+1}{2}, & \text{if } r \leq p < l \\ 0, & \text{if } p = l \end{cases}$$

Proof The proof is by induction on p . The validity of the result for $p = 1$ follows from Theorem 1. So, we assume its validity for p with $1 \leq p < l$.

Evidently, for $0 \leq r \leq p - 1$,

$$f^{p+1}(n) = \binom{l-p}{2}.$$

Now, for $p \leq r \leq l$, by Theorem 1,

$$\begin{aligned} & f\left(\binom{l-p+1}{2} + r - p\right) \\ &= \begin{cases} \binom{l-p}{2}, & \text{if } r - p = 0 \\ \binom{l-p}{2} + r - p - 1, & \text{if } r - p \neq 0. \end{cases} \end{aligned}$$

Hence, the result is also true for $p + 1$, completing the induction. \square

Now, given

$$n = \binom{m+1}{2} + r, \quad r \in \{0, 1, \dots, m\},$$

we define

$$s_p(n) = f^{p-1}(n) - f^p(n); \quad p = 1, 2, \dots, m. \quad (6)$$

Then, using Theorem 2, we get

$$n = s_1(n) + s_2(n) + \dots + s_m(n). \quad (7)$$

We thus see that the number of sizes, m , is just the (unique) number l satisfying the inequality (3), and hence m is given by

$$m = \text{trunc}(\sqrt{(8n+1)/4} - 1/2). \quad (8)$$

The sizes of the slices are given in Theorem 3 below, which follows directly from (6) and Theorem 2 after some algebraic manipulations.

THEOREM 3. For $p \in \{1, 2, \dots, m\}$,

$$s_p(n) = \begin{cases} m - p + 2, & \text{if } 1 \leq p \leq r \\ m - p + 1, & \text{if } r < p \leq m \end{cases}$$

3. CONCLUSIONS

The expression for $s_p(n)$, given in Theorem 3 above, coincides with that suggested by van de Liefvoort (1992). The above scheme guarantees that the smallest slice of size s_m can always be solved linearly; furthermore, Theorem 2, together with Proposition 2 of Newman-Wolfe (1986), shows that the smallest slice can also be moved using three pegs only, while the algorithm of van de Liefvoort uses all the four pegs for its movement.

We now have the following theorem, giving an expression for $M(n)$ in terms of the sizes of the optimal slices. Using Theorem 2, the equivalence of this with that of Liefvoort (his expression misses the additive term 1) can be established; however, from the computational point of view, that in Theorem 4 is more efficient.

THEOREM 4. Let, for some $m \in \{1, 2, \dots\}$,

$$\binom{m+1}{2} \leq n < \binom{m+2}{2}.$$

Then, with $s_i(n)$'s given by Theorem 3,

$$M(n) = \sum_{i=1}^m 2^{i-1} (2^{s_i(n)} - 1).$$

Proof. From (1) and (6), we have

$$\begin{aligned} M(n) &= 2M(f(n)) + 2^{s_1(n)} - 1 \\ &= 2(2M(f^2(n)) + 2^{s_2(n)} - 1) + (2^{s_1(n)} - 1), \end{aligned}$$

and continuing, we would ultimately get

$$M(n) = 2^m M(f^m(n)) + \sum_{i=1}^m 2^{i-1} (2^{s_i(n)} - 1).$$

Now, appealing to Theorem 2 and (2), the result follows. \square

The scheme (6) may be employed to find an optimal sequence of slices for the five-peg problem so that each slice can be solved with the four-peg algorithm. In his paper, van de Liefvoort (1992) reports that he failed to do so. The analysis of the five-peg problem will be done in a forthcoming paper.

REFERENCES

- Gedeon, T. D. (1992) The Reve's puzzle: an iterative solution produced by transformation. *Comp. J.*, **35**, 186–187.
- Hinz, A. M. (1989) An iterative algorithm for the Tower of Hanoi with four pegs. *Computing*, **42**, 133–140.
- van de Liefvoort, A. (1992) An iterative algorithm for the Reve's puzzle. *Comp. J.*, **35**, 91–92.
- Lu, X.-M. (1989) An iterative solution for the 4-peg towers of Hanoi. *Comp. J.*, **32**, 187–189.
- Newman-Wolfe, R. (1986) *Observations on Multi-Peg Towers of Hanoi*. Technical Report 187, Department of Computer Science, The University of Rochester, New York.
- Rohl, J. S. and Gedeon, T. D. (1986). The Reve's puzzle. *Comp. J.*, **29**, 187–188.