The N-Jugs and Water Problem

Thomas J. Pfaff

Math and Comp. Sci. Department

Ithaca College

1212 Williams Hall

Ithaca, NY 14850-7284

tpfaff@ithaca.edu

Max M. Tran

192 Santa Monica Drive

Columbus, OH 43213

maxminh@ccsol.net

February 24, 2005

1 Introduction

The n-jugs and water problems asks:

> Given n-jugs of size $L_1, L_2, \ldots, L_n$ liters and an unlimited supply of water, what integer amounts of water can be obtained and

stored in this collection of jugs?

Various forms of the case when $n = 2$ have appeared in numerous places including [2], the movie *Die Hard III*, and problem 10916 in the January 2002 *Monthly*. Also, we provided a group theoretic solution to the $n = 2$ problem in [3], but we only showed what amounts can be obtained and did not show what amounts could not be obtained. Here we will show, without using group theory, that if $d = \gcd(L_1, L_2, \ldots, L_n)$ then the only amounts that can be obtained are multiples of d between 0 and $L_1 + L_2 + \cdots + L_n$.

2 Example

We start with an example to illustrate the approach that we will use. Let $n = 3$, with $L_1 = 12$, $L_2 = 20$, and $L_3 = 30$. Note that

$$1 \times 12 + 1 \times 20 - 1 \times 30 = 2, \tag{1}$$

which we will use to show that with the given three jugs we can obtain a 2 liter amount of water. This equation tells us that we will bring water into the system by filling the 12-liter and 20-liter jug once. Meanwhile, water will have to leave the system by emptying the 30-liter jug once. Here is one way of doing this.

Fill the 12-liter jug and dump it into the 30-liter jug. Fill the 20-liter and top off the 30-liter jug, leaving 2 liters of water in the 20-liter jug. Now simply dump out the 30-liter jug. In this example, the gcd of any pair of jugs is greater than two, and hence two liters of water cannot be obtained from any two of the jugs. We will prove that with n jugs of size L_i liters, for

$1 \leq i \leq n$, we can obtain x liters of waters whenever we can find coefficients c_i such that

$$c_1 L_1 + \cdots + c_n L_n = x.$$

The algorithm is then to simply fill jug L_i, c_i times when $c_i > 0$, and empty jug L_i, c_i times when $c_i \leq 0$. In the next section we will show that such an algorithm can be performed with actual jugs and that x must be a multiple of the gcd of the jug sizes.

3 The Proof

Before preceding further we need to set out our axioms or rules of the game. We assume that there is an unlimited amount of water from a source. Whenever a jug is filled it is filled full, and jugs that are emptied are emptied completely. When water is poured from one jug to another, either the first jug is completely emptied, the second jug is completely filled, or both, whichever occurs first. In reality, these axioms simply insure that we know how much water is entering the system, exiting the system, or being transferred within the system. One would not partially fill or empty a jug since we would then have an unknown quantity of water entering or exiting the system, which would not help us obtain the given x liters of water.

Put $L = L_1 + L_2 + \cdots + L_n$ and Let $T = \{S = c_1 L_1 + \cdots + c_n L_n | c_i \in \mathbb{Z} \text{ and } 0 < S \leq L\}$. Thus, any amount S of water obtained in T comes from completely filling and emptying jugs. We now have two things to prove. First, we show that for all $S \in T$, the formula for S can be realized physically. Last, we prove that the set T is exactly multiples of d, the gcd of the jug sizes.

To see that we can execute the algorithm let $c_1L_1 + \cdots + c_nL_n \in T$, and without loss of generality, we can assume that $c_i > 0$ for $1 \le i \le p$ and that $c_i \le 0$ for $p < i \le n$. Now since $0 < c_1L_1 + \cdots + c_nL_n$ we get that

$$c_1L_1 + \cdots + c_pL_p > -(c_{p+1}L_{p+1} + \cdots + c_nL_n), \tag{2}$$

and from $c_1L_1 + \cdots + c_nL_n \le L$ we get that

$$c_1L_1 + \cdots + c_pL_p \le -(c_{p+1}L_{p+1} + \cdots + c_nL_n) + L. \tag{3}$$

We can rewrite (3) to get that

$$(c_1 - 1)L_1 + \cdots + (c_p - 1)L_p \le L_{p+1} + \cdots + L_n - (c_{p+1}L_{p+1} + \cdots + c_nL_n). \tag{4}$$

Now from (2) we see that more water enters into the system with jugs that need to be filled than leaves the system with jugs that need to be emptied, as expected. Whereas (4) shows us that, except for the final fill of a jug, as we fill the positive jugs, we can transfer and dump the water with the negative jugs and still have room to possibly store water in the negative jugs. The algorithm is now to simply fill the positive jugs and transfer the water to a negative jug and dump, starting with the L_1 and L_p jugs. We move to the next respective jug once we have filled and transferred water from the L_1 jug $c_1 - 1$ times or have dumped the L_p jug c_p times.

We now need to show that T is simply the set of multiples of $d = \gcd(L_1, \ldots, L_n)$ between 0 and L. By definition, $d|L_i$ for each i and so $d|(c_1L_1 + c_2L_2 + \cdots + c_nL_n)$ for any integers c_i, (see page 21 of [1]). Hence, every member of T is a multiple of d. Let $m = \min(T)$. By the division algorithm, $L_i = qm + r_i$ where $0 \le r_i < m$. If $r_i > 0$, then since $r_i = L_i - qm$ we would have $r_i \in T$ with $r_i < m$. Thus $r_i = 0$ and $m|L_i$ for each i. Since

m is an integer combination of the $\{L_i\}$ we have that $d|m$, and so $d \leq m$. On the other hand, by definition of d we have that $d \geq m$. Therefore, $d = m$ and so all multiples of d between 0 and L are in T.

What we have shown is that if we wish to obtain x liters of water, we first check to see if x is a multiple of d. If x is not a multiple of d then the amount cannot be obtained. If x is a multiple of d, then the Euclidean Algorithm provides us with an algorithm for finding the x liters of water.

Acknowledgement The authors would like to thank Ian Stewart for his comments on the first jug paper that led to this work. We would also like to thank the referee for improving our presentation.

References

[1] Burton, D., *Elementary Number Theory* 5th ed, McGraw Hill, New York, 2002.

[2] Mack, D., *The Unofficial IEEE Brainbuster Gamebook*, The Institute of Electrical and Electronics Engineers, New York, 1992.

[3] T. Pfaff and M. Tran, The Generalized Jug Problem, *Journal of Recreational Mathematics* 31-2 (2002-2003), 100-103.