Parabola Volume 52, Issue 1 (2016)

A new approach for solving the generalized 2 jugs decanting problem

Chavdar Lalov[1]

> *The Euclidean algorithm is the granddaddy of all algorithms, because it is the oldest nontrivial algorithm that has survived to the present day.*
>
> Donald Knuth, *The Art of Computer Programming, Vol. 2*, 1981

Decanting problems are widely used in recreational mathematics [1, p. 101], mathematics and informatics olympiads [2, 3], IQ and aptitude testing [4, p. 272], dynamic programming [5], analysis of algorithmic techniques in computer science [6], artificial intelligence [7, pp. 43–70], psychology [8] and more. In this article, we will examine the liquid measuring problem that begins with an unlimited supply of water and two unmarked jugs with given capacities m and n where m and n are integers.

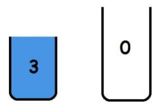
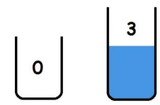
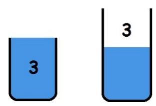
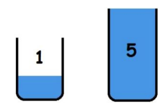
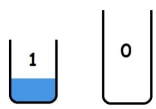
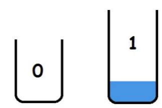
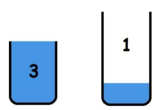
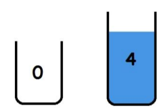
Consider the following operations each of which we will call a "step":

Fill a jug to its maximum capacity.
Pour the whole quantity of one jug into the other one if the liquid will fit.
Top up one jug with water from the other jug if there is enough liquid.
Empty a jug.

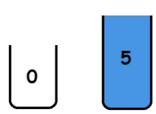
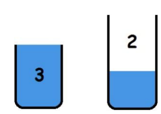
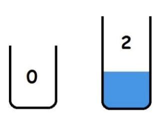
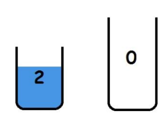
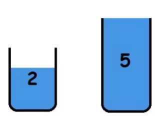
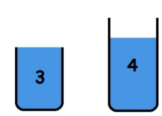
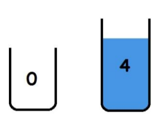
[1]Chavdar Lalov is a grade 9 student at Geo Milev Math High School, Pleven, Bulgaria.

Example 1: Let us start with a 2 water jug riddle seen in the movie *Die Hard 3*. This is so called *aquarius problem*: the heros have an unlimited supply of water (a fountain) and must measure exactly 4 L of water by using a 3 L jug and a 5 L jug. How can we solve this problem?

Solution 1

Steps	Jug states	Steps	Jug states
1. **Fill** the 3 L jug.		2. **Pour** the 3 L jug into the 5 L jug.	
3. **Fill** the 3 L jug.		4. **Top up** 5 L jug with the 3 L jug.	
5. **Empty** the 5 L jug.		6. **Pour** 1L from 3 L jug into 5 L jug.	
7. **Fill** the 3 L jug.		8. **Pour** the 3 L jug into the 5 L jug. In the 5 L jug we now have 4 L.	

Solution 2

Steps	Jug states	Steps	Jug states
1. **Fill** the 5 L jug.		2. **Top up** 3 L jug with the 5 L jug.	
3. **Empty** the 3 L jug.		4. **Pour** 2L in 5 L jug into 3 L jug.	
5. **Fill** the 5 L jug.		6. **Top up** the 3 L jug with 5 L jug.	
7. **Empty** the 3 L jug. We now have 4 L.			

We have two solutions with one and the same answer. This article will show you how to get both solutions or in two cases, which will be considered later, to find the shortest solution with the smallest number of steps possible. Let x and y be the variables which denote how many fillings from the water source or how many pourings we have done respectively with the m-liter and n-liter jug. Here, if x is positive, then we *filled* the m-liter jug x times; if x is negative, then we *poured* from the m-liter jug x times. The same is true for y. It turns out that we can make the two jugs problem algebraic by means of a Diophantine equation, namely $mx + ny = c$ [9, 11], where c will be the desired quantity of litres we want to obtain.

In the example above, we have $m = 5$, $n = 3$, $c = 4$ and the corresponding Diophantine equation $5x + 3y = 4$.

Quite obviously, neither of the capacities divides 4. Let us first find the solution of the following equation:

$$5x_1 + 3y_1 = \gcd(5, 3), \tag{1}$$

where $x_1, y_1 \in \mathbb{N}$ and $\gcd(5, 3)$ denotes the greatest common divisor of 5 and 3.

We use the extended Euclidean Algorithm [10] to compute the following steps:

a) The Euclidean Algorithm computes $g = \gcd(m, n)$.

b) The method of back-substitution yields integers $x, y \in \mathbb{Z}$ such that $mx + ny = g$.

The extended Euclidean algorithm allows us to find the solution of the Diophantine equation (1). In Table 1, by applying the extended Euclidean algorithm we get:

Table 1: The extended Euclidean algorithm applied for (1)

(a) Finding the $\gcd$ of m and n

$5 = 3 \times 1 + 2$
$3 = 2 \times 1 + 1$
$2 = 1 \times 2$

(b) Finding the values of x_1 and y_1

$1 = 3 - 2 \times 1$
$1 = 3 - (5 - 3 \times 1) \times 1$
$1 = 2 \times 3 - 5 \times 1$

Finding solution 1: From Table 1(a) and Table 1(b) we respectively get that

$$\gcd(5, 3) = 1 \quad \text{and} \quad 1 = 3 \times 2 - 5 \times 1.$$

Hence $\gcd(5, 3) = 1$ divides $c = 4$ and $5x + 3y = 4$ has a solution. Note that $c = \gcd(5, 3) \times 4 = 4$. Thus using the fact that

$$\gcd(5, 3) = 3y_1 + 5x_1 = 3 \times 2 - 5 \times 1 = 1,$$

and multiplying both sides by 4 we get

$$4(3 \times 2 - 5 \times 1) = 3(4 \times 2) - 5(1 \times 4) = 4,$$

and the solution of the Diophantine equation $5x + 3y = 4$ is $x = -4$ and $y = 8$.

We now examine whether we can decrease the absolute value of x and y. We decrease $|5 \times x| = 5 \times 4$ and $3 \times y = 3 \times 8$ by subtracting from both of them $\mathrm{lcm}(5,3) = 15 = 5 \times 3$, where $\mathrm{lcm}(5,3)$ denotes the least common multiple of 5 and 3. From 5×4 we can subtract 5×3 one time at the most, and from 3×8 also one time at the most. Therefore

$$(3 \times 8 - 3 \times 5) - (5 \times 4 - 5 \times 3) = 3 \times 3 - 5 \times 1 = 4.$$

In this way we see that $5x + 3y = 4$, where the absolute value of x and y are as small as possible (respectively $|-1| = 1$ and 3), but only in the case when y is positive and x is non-positive.

Finding solution 2: Now let us find the other solution where y is non-positive, x is positive and the absolute values of x and y are as small as possible. We subtract $\mathrm{lcm}(5,3)$ from 3×3 and 5×1 and this gives $3 \times 3 - 3 \times 5 = -3 \times 2$ and $5 \times 1 - 5 \times 3 = -5 \times 2$. Consequently we have

$$5 \times 2 - 3 \times 2 = 10 - 6 = 4,$$

and thus we find a second solution $x = 2$ and $y = -2$.

Generalization for arbitrary m and n

The aim of our method is to find a quick way to pour the jugs, i.e. to find the smallest values of $x, y \in \mathbb{N}$ such that

$$mx + ny = c\,. \tag{2}$$

Using the extended Euclidean algorithm we can find the solution of the equation

$$mx_1 + ny_1 = \gcd(m,n)\,, \tag{3}$$

where $x_1, y_1 \in \mathbb{N}$. Obviously either x_1 or y_1 is non-positive; assume it to be y_1. Let c represent the quantity (in litres) we want to get. This quantity is smaller than or equal to the maximum between m and n otherwise if c is larger than both m and n then the desired quantity does not fit in one jug .

We want to find the solution of (2). By Bezout's identity, a solution exists only if $\gcd(m,n)$ divides c. Therefore we can write $c = \gcd(m,n) \times k$ where $k \in \mathbb{N}$, and multiplying both sides of (3) by k gives

$$(mx_1 + ny_1)k = \gcd(m,n)k = c\,.$$

Therefore

$$(kx_1)m + (ky_1)n = mx + ny = c\,.$$

We know that y is non-positive. Hence $mx + ny$ may be represented as $mx + n|y|$, and it is clear that $mx > n|y|$. It is obvious that the minimal solution is obtained when the sum of x and $|y|$ is minimal, or else the quantity would be obtained at least twice during the steps. It is easy but a little tedious to prove that the minimum of the sum of x and $|y|$ occurs when x and $|y|$ are minimal.

We now examine the case whether the value of x and $|y|$ can be decreased while their values remain non-negative. Thus we want to subtract a number divisible by m from both $x = (kx_1)m$ and $y = |ky_1|n$ to reduce x, and by analogy a number divisible by n from $x = (kx_1)m$ and $y = |ky_1|n$ to reduce y. To do so we subtract $\operatorname{lcm}(m, n)$ from these quantities as many times as it is possible.

Additionally, it turns out that $\operatorname{lcm}(m, n)$ can be subtracted as many times from $x = (kx_1)m$ than from $y = |ky_1|n$ (unless n divides c). The lengthy but simple proof is left to the reader.

Finally an illustration of the method for some higher volumes is given below.

Example 2: We have two jugs of 7777 L and 1239 L and want to measure 14 L.

First, using the Extended Euclidean algorithm (see Table 2), we find the solution to

$$7777x_1 + 1239y_1 = \gcd(7777, 1239). \tag{4}$$

Table 2: The extended Euclidean algorithm applied for (4)

(a) Finding the $\gcd$ of m and n

$7777 = 1239 \times 6 + 343$
$1239 = 3 \times 343 + 210$
$343 = 210 \times 1 + 133$
$210 = 133 \times 1 + 77$
$133 = 77 \times 1 + 56$
$77 = 56 \times 1 + 21$
$56 = 21 \times 2 + 14$
$21 = 14 \times 1 + 7$
$14 = 7 \times 2$

(b) Finding the values of x_1 and y_1

$7 = 21 \times 1 - 1 \times 14$
$7 = 21 \times 3 - 56 \times 1$
$7 = 3 \times 77 - 4 \times 56$
$7 = 7 \times 77 - 4 \times 133$
$7 = 7 \times 210 - 11 \times 133$
$7 = 18 \times 210 - 11 \times 343$
$7 = 18 \times 1239 - 65 \times 343$
$7 = 408 \times 1239 - 65 \times 7777$

Finding solution 1: From Table 2(a) and Table 2(b) we respectively get that

$$\gcd(7777, 1239) = 7 \quad \text{and} \quad 7 = 408 \times 1239 - 65 \times 7777\,.$$

Since 7 divides 14 we have a solution to $1239y + 7777x = 14$. Multiply the equation $7 = 408 \times 1239 - 65 \times 7777$ by 2 to get the following result:

$$816 \times 1239 - 130 \times 7777 = 14\,.$$

We want to find the solution where the absolute values of x and y are minimal using $\operatorname{lcm}(7777, 1239)$. We know that

$$\operatorname{lcm}(7777, 1239) = 7777 \times 177 = 1239 \times 1111 = 1376529\,.$$

However here $\operatorname{lcm}(7777, 1239)$ cannot be subtracted from 816×1239 and 130×7777 without changing the sign of x and y. Thus the first solution is $816 \times 1239 - 130 \times 7777 = 14$.

In conclusion, we get $14\,\mathrm{L}$ by filling the $1239\,\mathrm{L}$ jug 816 times, repeatedly pouring its contents into the $7777\,\mathrm{L}$ jug and, in the process, emptying out the $7777\,\mathrm{L}$ jug 130 times.

Finding solution 2: We subtract $\mathrm{lcm}(7777, 1239)$ from both 816×1239 and 130×7777, and we get

$$47 \times 7777 - 295 \times 1239 = 14\,,$$

which is the other desired solution. In conclusion, we get $14\,\mathrm{L}$ by filling the $7777\,\mathrm{L}$ jug 47 times, repeatedly pouring its contents into the $1239\,\mathrm{L}$ jug and, in the process, emptying out the $1239\,\mathrm{L}$ jug 295 times.

Consequently, we have found both solutions to the problem.

Conclusion

The introduced problem is one that has been studied from ancient to modern times. We have shown how to find both of the solutions or the quickest one of them. At the same time the method gives us opportunity to find the solution when we have to deal with decanting jugs with high volume. Moreover, it helps us to avoid retracing our steps or in other ways making wasteful steps. The current project could be extended by studying the decanting problem when using more than two jugs. This is left for the interested reader to explore.

References

[1] B. Averbach and O. Chein, *Problem Solving Through Recreational Mathematics*, Dover Publications, 2000.

[2] *Manhattan Mathematical Olympiad 2001*, Grades 5–6, Problem 4, Retrieved from https://www.math.ksu.edu/~soibel/Olympiad/2001-5-6.pdf [Assesed 21 February 16]

[3] *The 2010 British Informatics Olympiad*, Question 3, Retrieved from http://www.olympiad.org.uk/papers/2010/bio/bio-10-exam.pdf [Assesed 21 February 16]

[4] R.J. Sternberg, *Handbook of Human Intelligence*, Cambridge University Press, 2000.

[5] *The Classic Problem - Water*, Retrieved from http://www.programering.com/a/MjMxcTNwATk.html [Accessed 21 February 16]

[6] *Making and Following Algorithms*, Retrieved from http://www.oxfordmathcenter.com/drupal7/node/409 [Accessed 21 February 16]

[7] C. Thornton, B. Du Boulay, *Artificial Intelligence Through Search*, Springer, 2012.

[8] M.E. Atwood, M.E. Masson and P.G. Polson, Further explorations with a process model for water jug problems, *Memory and Cognition* **8** (1980), 182–192.

[9] S.S. Epp, *Discrete Mathematics: Introduction to Mathematical Reasoning*, Cengage Learning, 2011.

[10] S.P. Glasby, Extended Euclid's algorithm via backward recurrence relations, *Mathematics Magazine* **72**(3) (1999), 228–230.

[11] Y.K. Man, Solving the water jugs problem by an integer sequence approach, *International Journal of Mathematical Education in Science and Technology* **43** (2012), 109–113.

[12] H. Reiter, *Mathzoom Academy Summer 2014, Decanting Problems*, Retrieved from http://math2.uncc.edu/~hbreiter/Zoom/MZDecanting.pdf [Assesed 21 February 16]