An Open Pouring Problem

Fabian Frei
Department of Computer Science, ETH Zürich, Switzerland
fabian.frei@inf.ethz.ch

Peter Rossmanith
Department of Computer Science, RWTH Aachen, Germany
rossmani@cs.rwth-aachen.de

David Wehner
Department of Computer Science, ETH Zürich, Switzerland
david.wehner@inf.ethz.ch

Abstract

We analyze a little riddle that has challenged mathematicians for half a century.

Imagine three clubs catering to people with some niche interest. Everyone willing to join a club has done so and nobody new will pick up this eccentric hobby for the foreseeable future, thus the mutually exclusive clubs compete for a common constituency. Members are highly invested in their chosen club; only a targeted campaign plus prolonged personal persuasion can convince them to consider switching. Even then, they will never be enticed into a bigger group as they naturally pride themselves in avoiding the mainstream. Therefore each club occasionally starts a campaign against a larger competitor and sends its own members out on a recommendation program. Each will win one person over; the small club can thus effectively double its own numbers at the larger one's expense.

Is there always a risk for one club to wind up with zero members, forcing it out of business? If so, how many campaign cycles will this take?

2012 ACM Subject Classification Theory of computation → Theory and algorithms for application domains; Theory of computation → Representations of games and their complexity

Keywords and phrases Pitcher Pouring Problem, Water Jug Riddle, Water Bucket Problem, Vessel Puzzle, Complexity, Die Hard

Digital Object Identifier 10.4230/LIPIcs.FUN.2021.14

Acknowledgements We would like to thank the anonymous reviewers who carefully read this paper for their detailed feedback.

1 The Same Old Pouring Problem Again[1]

Almost anyone who is even remotely fond of logical puzzles and many others have heard of and even solved the following problem:

> *Given two pitchers of three and five ounces capacity and an infinite water supply, can you precisely measure four ounces?*

This already popular pitcher-pouring problem has gained increased prominence when it was featured in the third installment of the Die Hard movie series, released in 1995. The two protagonists John and Zeus are forced to figure out the solution to the problem within five minutes to defuse a bomb. Frantically discussing the problem in detail, they eventually succeed and prevent the explosion with mere seconds left, as expected.

[1] Please patiently pardon particularly peculiar pour puns.

An outpouring of papers discussing different aspects of the problem ensued – considering the natural generalization to arbitrary capacities of the two pitchers, of course. It also has become somewhat of a pet problem in Artificial Intelligence [8, 6]. The focus was often on didactical aspects of the problem as the solution is rather simple from a mathematical standpoint: The number of liters that can be measured using two given pitchers are exactly the greatest common divisor of the two capacities and its multiples. The complexity of the problem – that is, the number of steps required to solve all instances of the problem with a given total capacity – was first considered and immediately shown to be linear directly following the movie release in 1995 as well [5].

Fortunately, there is a far more challenging pouring problem that is still open.

2 Our Problem: Significantly Less Pouring

The fifth problem of the fifth All-Union round of the Soviet Mathematics Olympiad, held in Riga in 1971, reads as follows:

> В три сосуда налито по целому числу литров воды. В любой сосуд разрешается перелить столько воды, сколько в нем уже содержится, из любого другого сосуда. Докажите, что несколькими такими переливаниями можно освободить один из сосудов. (Сосуды достаточно велики: каждый может вместить всю имеющуюся воду.) [7]

> *An integer number of liters of water has been poured into each of three vessels. It is allowed to pour into each vessel as much water as it already contains from an arbitrary other vessel. Prove that several such pourings can empty one of the vessels. (The vessels are sufficiently large: Each one can contain the entire available water.)*

Two small clarifications might be in order. First, each of the three vessels may contain a different natural number of liters of course; the puzzle would be trivial otherwise. Second, it is only implicit that we can never pour water out of a vessel that contains less water than the receiving one. See Figure 1 for an illustrating example with a simple instance and its optimal solution.

The exam designer clearly made an effort to keep the task from being too abstract by casting it into this vessel form. Nevertheless, coming up with a way to perform such magical pouring steps that allow us to double a vessel's content – without additional materials that would render the entire enterprise obsolete – seems to be the hardest challenge here by far.

However, even when freeing ourselves from the pour setting, it remains a tough task to find any natural situation where the described situation might arise. While we gave it our best try in the abstract, the issue is usually skirted altogether, as we will see.

The puzzle made an honoring appearance as the final task in the 54th iteration of the William Lowell Putnam Mathematical Competition. The organizers of the most prestigious mathematical competition worldwide opted for a purely abstract description:

> *Let S be a set of three, not necessarily distinct, positive integers. Show that one can transform S into a set containing 0 by a finite number of applications of the following rule: Select two of the three integers, say x and y, where $x \leq y$ and replace them with $2x$ and $y - x$.* [1]

Two decades later, when the time-proven problem was presented as a challenge to IBM researchers, their puzzlemaster embedded it in a contrived betting game [2]. Most recently, the task took yet another, now overtly magical guise in Germany's 38th National Computer Science Competition [3]. This time around, the participants were not asked to solve the problem mathematically, but had to write a program solving it optimally instead.

Now, this problem was not only foisted upon defenseless exam takers; it has also been included into a carefully curated anthology of mathematical riddles aimed at every avid enigma aficionado [10, 9, 11]. In "Mathematical Puzzles: A Connoisseur's Collection," the puzzle-pondering professor Peter Winkler, a well-respected mathematician and computer scientist, presents the problem in its original form with three water buckets. He goes on to describe his solution to the problem, which might well have been the intended one. His approach guarantees that an initial configuration with n liters in total can be turned into one with an empty bucket in at most $\mathcal{O}(n^2)$ steps.

This result was independently improved upon by two persons with whom Peter Winkler had been sharing the puzzle, Svante Janson from Uppsala University and Garth Payne from Pennsylvania State University. They both described an algorithm that can empty one of the buckets in at most $\mathcal{O}(n \log n)$ steps. Winkler concludes: *"As far as I know, no one knows even approximately how many steps are required for this problem."* [9]

The German translation of the book, published in 2008, adds an optimistic conjecture by Michael H. Albert that the minimal number of steps is far lower, namely $\Theta((\log n)^2)$.

3 Our Contribution

In this section, we first improve in Subsection 3.2 the known upper bound from the linearithmic $\mathcal{O}(n \cdot \log n)$ down to $\mathcal{O}((\log n)^2)$, matching Albert's conjecture. In Subsection 3.2, we then go on to give experimental evidence that, on the one hand, exhibits the very peculiar and mathematically interesting behavior of this problem and, on the other hand, strongly suggests that even Albert's conjecture is too pessimistic still: The required number of steps is far more likely to be $\Theta(\log n)$, which we posit as our improved conjecture. In Subsection 3.3, we prove a lower bound that not only matches our conjecture asymptotically but in fact perfectly fits the experimental data for infinitely many n that we analyze more closely in Subsection 3.4. Note that our results leave open a good gap of a single logarithmic factor, lest Winkler's puzzling problem be spoiled entirely for the reader.

We briefly restate the problem in the notation that will be used in our proofs.

> Let $(a, b, c) \in \mathbb{N}^3$ be a triple of nonnegative integers. We are allowed to perform the following modification step on any triple: Pick from it any two numbers x and y such that $x \leq y$ and then replace them by $2x$ and $y - x$, respectively.
> For any $n \in \mathbb{N}$, denote by $N(n)$ the minimum number of such steps that allows us to transform any given triple $(a, b, c) \in \mathbb{N}^3$ satisfying $a+b+c = n$ into a triple containing a zero. Prove good upper and lower bounds on $N(n)$.

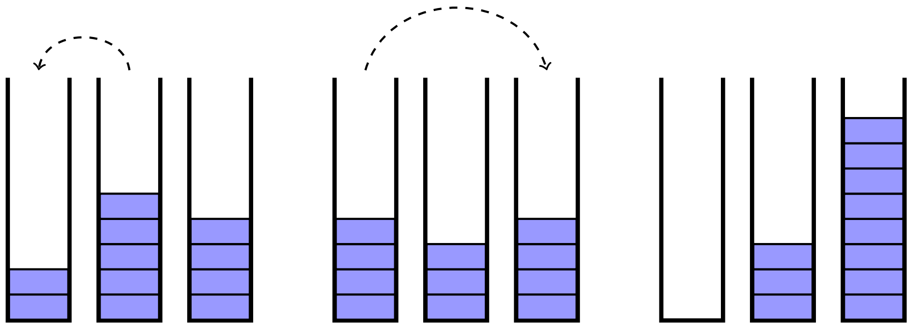
3.1 Upper Bound

We directly state and then prove our upper bound.

▶ **Theorem 1.** *The optimal number of steps required to solve any instance with a total liter count of n is bounded by $N(n) \leq (\log n)^2$.*[2]

Proof. We describe an algorithm that transforms any given configuration $(a, b, c) \in \mathbb{N}^3$ into one containing a zero in at most $(\log n)^2$ steps, where $n = a + b + c$. Our algorithm works in rounds. We may assume without loss of generality that every round starts with

[2] Throughout this paper, log denotes the logarithm to base 2.



(a) Initial Configuration. **(b)** One pouring later. **(c)** Problem solved.

Figure 1 Optimal solution for the initial instance $(2, 5, 4)$. The total number of liters is $n = 11$, the required number of pouring steps is $N(n) = 2$.

a configuration (a, b, c) that is ordered such that $0 < a \leq b \leq c$. In every round, it will transform the configuration (a, b, c) into a new configuration (a', b', c') satisfying $a' \leq a/2$, using a series of at most $\log n + 1$ steps. As any configuration in ascending order, the initial configuration satisfies $a \leq n/3$. It is thus guaranteed that after at most $\log n - 1$ rounds we reach a configuration whose smallest number is at most $n/3 \cdot 2^{1-\log n} = 2/3$, which just means it is zero, as required. We note that Svante's algorithm as described by Winkler [9] is structured in rounds as well. Instead of halving the number a in each round, it only guarantees a decrease by at least 1, however. This crucial difference allows us to improve the upper bound from $\mathcal{O}(n \cdot \log n)$ to $\mathcal{O}((\log n)^2)$.

We now describe a single round that starts with a configuration (a, b, c) satisfying $0 < a \leq b \leq c$. Let $r := b/a$ be the ratio between the two smallest numbers. We round this ratio both ways and denote the resulting integers by $p := \lfloor r \rfloor$ and $q := \lceil r \rceil$. Let $p_k \ldots p_0$ and $q_\ell \ldots q_0$ be the minimal binary representations of p and q, respectively; that is, we have $p_0, \ldots, p_{k-1}, q_0, \ldots q_{\ell-1} \in \{0, 1\}$ and $p_k = q_\ell = 1$ for $k = \lfloor \log p \rfloor$ and $\ell = \lfloor \log q \rfloor$ with $\sum_{i=0}^{k} p_i 2^i = p$ and $\sum_{i=0}^{\ell} q_i 2^i = q$. Note that $0 \leq b - pa < a$ and $0 \leq qa - b < a$. This implies that $\min\{b - pa, qa - b\} \leq a/2$ since $(b - pa) + (qa - b) = (q - p)a \leq a$. We can thus consider the following two, potentially overlapping cases.

Case 1: Assume first that $b - pa \leq a/2$. In this case we perform $k + 1$ steps for $i = 0, \ldots, k$ that will always double what was initially the smallest number a in the triple. To do this, the algorithm has to subtract first a, then $2a$, and generally $2^i a$ from one of the other two numbers in the triple. We use p to decide which one: If $p_i = 1$, we subtract $2^i a$ from the second number, which initially is b; otherwise we have $p_i = 0$ and subtract from the third number, which is c initially. After performing these $k + 1$ steps, the second number and third number will be $b - \sum_{i=0}^{k} p_i 2^i a = b - pa \leq a/2$ and $c - \sum_{i=0}^{k} (1 - p_i) 2^i a$, respectively. We have to prove that both the b and c of the initial configuration are sufficiently large so as to never become negative and thus make all steps in this round valid. For b, we can simply use our general observation $b - pa \geq 0$. For c, we have

$$c - \sum_{i=0}^{k} (1 - p_i) 2^i a = c - \sum_{i=0}^{k-1} (1 - p_i) 2^i a = c - \left(\sum_{i=0}^{k-1} 2^i - \sum_{i=0}^{k-1} p_i 2^i \right) a \geq c - 2^k a \geq 0,$$

where the last inequality follows from $k = \lfloor \log p \rfloor$ and $a \leq c$. We conclude that this round is feasible and results in a triple whose smallest number is at most $b - pa \leq a/2$, as required. The number of steps in this round is exactly $k + 1 = \lfloor \log p \rfloor + 1 \leq \lfloor \log q \rfloor + 1$.

Case 2: We now assume $qa - b < a/2$. In this case, we first perform the following ℓ steps for $i = 0, \dots, \ell - 1$: We always double the first number a in the triple, as we did in the first case, but now we will be subtracting the necessary amount from the second number, the initial b, if $q_i = 1$ and from the third number, the initial c, if $q_i = 0$. After these ℓ steps, the first number of the resulting triple will be $2^\ell a$, the second one $b - \sum_{i=0}^{\ell-1} q_i 2^i a = b - (q - 2^\ell)a$ and the third one $c - \sum_{i=0}^{\ell-1}(1 - q_i)2^i a$. Again, we have to prove that these ℓ steps are in fact possible by showing that b and c are large enough. This is the case because, on the one hand, we have $q \leq p + 1$ and thus $b - (q - 2^\ell)a \geq b - (q - 2^0)a \geq b - pa \geq 0$ and, on the other hand, we have

$$c - \sum_{i=0}^{\ell-1}(1 - q_i)2^i a \geq c - \sum_{i=0}^{\ell-1} 2^i a = b - (2^\ell - 1)a \geq b - (q - 1)a \geq b - pa \geq 0.$$

We now perform the last, $(\ell + 1)$st step of this round. It doubles the second number $b - (q - 2^\ell)a = b - qa + 2^\ell a$ and subtracts the corresponding amount from the first number, which currently is $2^\ell a$. This step is valid since $2^\ell a - (b - qa + 2^\ell a) = qa - b \geq 0$. The round ends with this step and the first number of the triple is now $qa - b < a/2$. The number of steps in this round is precisely $\ell + 1 = \lfloor \log q \rfloor + 1$.

We have shown for both cases how to perform a valid round of at most $\lfloor \log q \rfloor + 1 \leq \lfloor \log(n/1) \rfloor + 1 \leq 1 + \log n$ steps that result in a triple whose smallest number is at most $\lfloor a/2 \rfloor$. As already mentioned, it now suffices to iterate this entire process for at most $(\log n) - 1$ rounds and we end up with a final configuration whose smallest number is 0. The total number of steps over all rounds is thus at most $(\log(n) - 1) \cdot (\log(n) + 1) \leq (\log n)^2$, which proves the theorem. ◀

3.2 Experimental Evidence

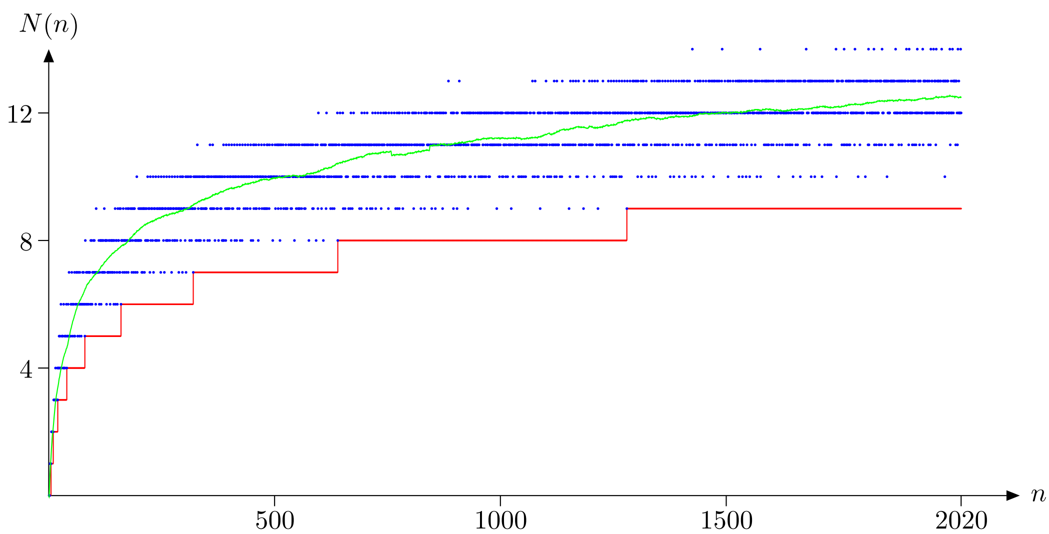


Figure 2 For any blue point, the ordinate indicates the minimum number of pouring steps required in the worst case for a starting configuration (a, b, c) with $n = a + b + c$, where n is given by the abscissa. The green line shows for each n the average of the 85 blue points $(n - 42, \dots, n + 42)$. The red line plots our lower bound derived in Subsection 3.3.

Table 1 All smallest hard instances for the listed minimum number of steps, that is, all worst-case instances for the smallest n yielding a new step number $N(n)$ for n up to 2020. The values of n are also found as sequence A256001 in the online encyclopedia of integer sequences [4].

$N(n)$	$n = a + b + c$	(a, b, c)
1	3	$(1, 1, 1)$
2	6	$(1, 2, 3)$
3	11	$(1, 4, 6)$
4	15	$(4, 5, 6)$, $(3, 4, 8)$, $(2, 5, 8)$
5	23	$(3, 8, 12)$
6	27	$(5, 9, 13)$
7	45	$(4, 15, 26)$
8	81	$(8, 27, 46)$
9	105	$(27, 35, 43)$, $(8, 35, 62)$, $(8, 27, 70)$
10	195	$(57, 65, 73)$, $(8, 78, 109)$, $(4, 78, 113)$, $(8, 73, 114)$, $(8, 65, 122)$, $(4, 66, 125)$, $(8, 57, 130)$, $(4, 33, 158)$
11	329	$(4, 130, 195)$
12	597	$(175, 199, 223)$
13	885	$(101, 295, 489)$
14	1425	$(206, 475, 744)$

In order to develop a proper intuition of the behavior of the step complexity $N(n)$, we wrote a program that calculates $N(n)$ for any given n. It does so by exhaustive enumeration of all instances and then trying out all feasible solutions. We show our findings in Figure 2. The blue points plot $N(n)$, the step number required for a worst-case triple whose numbers sum up to n, against this total for n from 0 up to 2020.

Across the entire spectrum, we observe erratic jumps up and down. The values where these jumps occur do not seem to follow any discernible pattern, however. Despite their best efforts, the authors were indeed unable to detect any stable structure, except for a small detail that we will discuss later on.

From a global perspective, taking a step back and squinting a little bit, a clear consistent picture emerges out of the confusing micro-behavior: The bulk of the values clearly follows a logarithmic line; we can approximate it by the green line, which plots the average value of $N(n)$ across the interval $[n - 42, n + 42]$. The oscillations of the blue points around the imagined center of gravity begin very small, but grow in amplitude with increasing n. The plotted pairs (n, N) where the amplitude first increases over the previous threshold are the following: From $(15, 4)$ to $(16, 3)$, from $(26, 4)$ to $(27, 6)$, from $(105, 9)$ to $(112, 6)$, and from $(885, 12)$ to $(896, 8)$. The gaps from one threshold to the next appear to be growing exponentially, but again the data is too noisy to deduce any rule.

Another point of interest might be the first values for n at which $N(n)$ spikes up to a new height for the first time. In order to describe the optimal monotone upper bound for this problem, we would need to understand these values. However, they do not exhibit any clear pattern either, besides an approximately exponential growth. The values of n up to 2020 for which $N(n)$ reaches a previously unattained value and the corresponding worst-case instances are displayed in Table 1.

Once more, neither an underlying nor an overarching pattern was to be found, neither in the instances themselves nor in the optimal solutions' step sequences.

To address the complementary question about the optimal monotone *lower* bound, we should at least know what the largest values n keeping $N(n)$ at any fixed value are. Seeing how we have been pouring out all the intractabilities of our problem on the reader, it might be surprising that we can in fact give a quite concise answer to this last question: For any $\ell \in \mathbb{N} \setminus \{0\}$, the largest n to yield $N(n) = \ell$ is $n = 5 \cdot 2^\ell$ and the instance

$$\left(\left\lfloor \frac{5}{3} \cdot 2^\ell \right\rceil - 1, \left\lfloor \frac{5}{3} \cdot 2^\ell \right\rceil, \left\lceil \frac{5}{3} \cdot 2^\ell \right\rceil + 1\right),$$

where by $\lfloor x \rceil$ we denote x rounded to the nearest integer, emerged as the corresponding unique worst-case instance. We will investigate these instances closer in Subsection 3.4.

3.3 Lower Bound

We finally prove our lower bound and show how well it matches our experimental data.

▶ **Theorem 2.** *The number of steps for solving a worst-case instance with a total liter count of n is at least $\lceil \log((n+1)/5) \rceil = \Omega(\log n)$.*

Proof. Let $t \coloneqq \frac{n}{3}$ and consider the following configuration (a, b, c) depending on the remainder of n modulo 3:

$$(a, b, c) = \begin{cases} (t-1, t, t+1), & \text{if } t - \lfloor t \rfloor = 0\,, \text{ i.e., } n \equiv 0 \pmod 3 \\ (t - \frac{4}{3}, t - \frac{1}{3}, t + \frac{5}{3}), & \text{if } t - \lfloor t \rfloor = \frac{1}{3}, \text{ i.e., } n \equiv 1 \pmod 3 \\ (t - \frac{5}{3}, t + \frac{1}{3}, t + \frac{4}{3}), & \text{if } t - \lfloor t \rfloor = \frac{2}{3}, \text{ i.e., } n \equiv 2 \pmod 3. \end{cases}$$

We can write this down in general as $(t+d_1, t+d_2, t+d_3)$ with $d_1 < d_2 < d_3$. After one step, we have either $(2t+2d_1, d_2-d_1, t+d_3)$ or $(2t+2d_1, t+d_2, d_3-d_1)$ or $(t+d_1, 2t+2d_2, d_3-d_2)$.

By re-ordering, all of these configurations can be written as $(x_1, t + y_1, 2t + z_1)$. We will generally write the configuration after i steps as $(x_i, t + y_i, 2t + z_i)$. Let u_i, v_i, and w_i be the absolute values of x_i, y_i, and z_i in ascending order, that is, we always have $\{u_i, v_i, w_i\} = \{|x_i|, |y_i|, |z_i|\}$ and $u_i \le v_i \le w_i$. For $i = 1$, we can directly verify that $w_i \le 5/3 \cdot 2$ and $v_i \le 5/3 \cdot 2 - 1/3$. In general, we observe that the largest absolute value after step i, namely w_i, can be at most double of what the previously largest absolute value was; we have $w_i \le 2w_{i-1}$. Moreover, the second largest absolute value v_i can be at most the sum of the two largest absolute values in the preceding step; we have $v_i \le v_{i-1} + w_{i-1}$. By induction, we therefore obtain $u_i \le v_i \le w_i \le 5/3 \cdot 2^i$ and $v_i \le 5/3 \cdot 2^i - 1/3$ for the absolute values after the ith step. It immediately follows that

$$v_i + w_i \le 5/3 \cdot 2^{i+1} - 1/3. \tag{1}$$

Clearly, as long as $v_i + w_i < t$, no two of the three numbers can add up to t and thus a, b, c cannot be equal. However, the only way to reach a configuration that contains a zero is from a configuration with two equal numbers. Thus, $v_i + w_i \ge t = n/3$ is a necessary condition for $(x_i, t + y_i, 2t + z_i)$, the configuration after step i, to contain two equal numbers. Using the bound (1) derived above, we conclude that this condition will not be met as long as the following two equivalent inequalities remain true:

$$\frac{5}{3} \cdot 2^{i+1} - \frac{1}{3} < \frac{n}{3} \qquad \Longleftrightarrow \qquad i + 1 < \log \frac{n+1}{5}.$$

Consequently, any step number k that affords us just a chance of ending up with two equal numbers has to satisfy $k + 1 \ge \log((n+1)/5)$. The number of steps k being an integer, we can improve this to $k \ge \lceil \log((n+1)/5) \rceil - 1$. Only after at least k steps we might have two equal number appearing in our configuration. One additional step involving these two numbers is then required to produce a zero. Therefore, the minimum number of steps is at least $\lceil \log((n+1)/5) \rceil = \Omega(\log n)$. ◀

We would now like to present evidence that this bound is in fact optimal for infinitely many values of n in the following section.

3.4 Solving Hard Instances Optimally

We complement the lower bound derived in Subsection 3.3 with an analysis of all instances of the form

$$\left(\left\lfloor \frac{5}{3} \cdot 2^{\ell} \right\rfloor - 1, \left\lfloor \frac{5}{3} \cdot 2^{\ell} \right\rceil, \left\lceil \frac{5}{3} \cdot 2^{\ell} \right\rceil + 1\right),$$

where $n = 5 \cdot 2^{\ell}$ for any natural number ℓ.

For these instances, our lower bound evaluates to $\lceil \log(2^{\ell} + 1/5) \rceil = \ell + 1$. (Note that these instances have the form $(t - 4/3, t - 1/3, t + 5/3)$ and $(t - 5/3, t + 1/3, t + 4/3)$ for even and odd ℓ, respectively.) We will now show that $\ell + 1$ steps are indeed sufficient for solving these instances.

For $\ell = 0$ and $\ell = 1$, $N(n) \geq \ell + 1$ is easily verified by checking all possibilities. Now let $\ell \geq 2$ and $n = 5 \cdot 2^{\ell}$.

Case 1: Assume that ℓ is odd. Let $k = (\ell - 3)/2$. Using binary representation, we can now represent the numbers in our hard instance (a, b, c) as

$$\begin{aligned} a &= 11(01)^k 00_2, \\ b &= 11(01)^k 01_2, \text{ and} \\ c &= 11(01)^k 11_2. \end{aligned}$$

It is easy to check that these numbers sum up to $n = 101(0)_2^{\ell}$. We start by transferring from the last number to the middle one, yielding $(a, 2b, c - b) = (a, 2b, 2) = (11(01)^k 00_2, 11(01)^k 010, 10_2)$. We then alternately subtract the last number from the second and the first, beginning with the second, for $2k + 2$ steps in total. Finally, we subtract the last number from the first one again. The first number will then be zero after a total of $1 + (2k + 2) + 1 = 2k + 4 = \ell + 1$ steps.

Case 2: Assume that ℓ is even. Let $k = (\ell - 4)/2$. Using binary representation again, we have

$$\begin{aligned} a &= 11(01)^k 001_2, \\ b &= 11(01)^k 011_2, \text{ and} \\ c &= 11(01)^k 100_2. \end{aligned}$$

Now, we first go from (a, b, c) to $(2a, b - a, c) = (2a, 2, c) = (11(01)^k 0010_2), 10_2, 11(01)^k 100_2)$. We then subtract the middle number from the first and then from the last. Now we begin to subtract the middle number alternately from the last and the first, beginning with the last, for $2k + 1$ steps in total. Finally, we subtract once more the middle number from the last number. This will result in the last number becoming zero after a total of $1 + 2 + (2k + 1) + 1 = 2k + 5 = \ell + 1$ steps.

We have thus shown $\log(2n/5)$ to be the optimal bound for infinitely many hard instances.

4 L'Art Pour l'Art

We hope to have sparked in the reader an unquenchable enthusiasm for the presented pouring-pot problem, prompting a perplexing pot-pourri of pertinent papers and perceptive proofs from our prodigious puzzle partners.

References

1 URL: `https://kskedlaya.org/putnam-archive/1993.pdf`.
2 URL: `https://www.research.ibm.com/haifa/ponderthis/challenges/May2015.html`.
3 URL: `https://bwinf.de/fileadmin/bundeswettbewerb/38/BwInf38-Aufgabenblatt.pdf`.
4 URL: `https://oeis.org/A256001`.
5 Paolo Boldi, Massimo Santini, and Sebastiano Vigna. Measuring with jugs. *Theor. Comput. Sci.*, 282(2):259–270, 2002.
6 Yiu-Kwong Man. A non-heuristic approach to the general two water jugs problem. *Theor. Comput. Sci.*, (10):904–908, 2013.
7 Николай Борисович Васильев (Nikolaj Borisovich Vasil'ev) and Александр Александрович Егоров (Aleksandr Aleksandrovich Egorov). *Задачи всесоюзных математических олимпиад (Zadachi Vsesojuznyh Matematicheskih Olimpiad, Problems of the All-Union Mathematical Olympiads)*. Наука (Nauka), 1988.
8 Glânffrwd P. Thomas. The water jugs problem: Solutions from artificial intelligence and mathematical viewpoints. *Mathematics in School*, 24(5):34–37, 1995.
9 Peter Winkler. *Mathematical Puzzles: A Connoisseur's Collection*. A K Peters, 2004.
10 Peter Winkler. Five algorithmic puzzles. In *Tribute to a Mathemagician*, pages 109–118. A K Peters, 2005.
11 Peter Winkler. *Mathematische Rätsel für Liebhaber*. Springer, 2008.