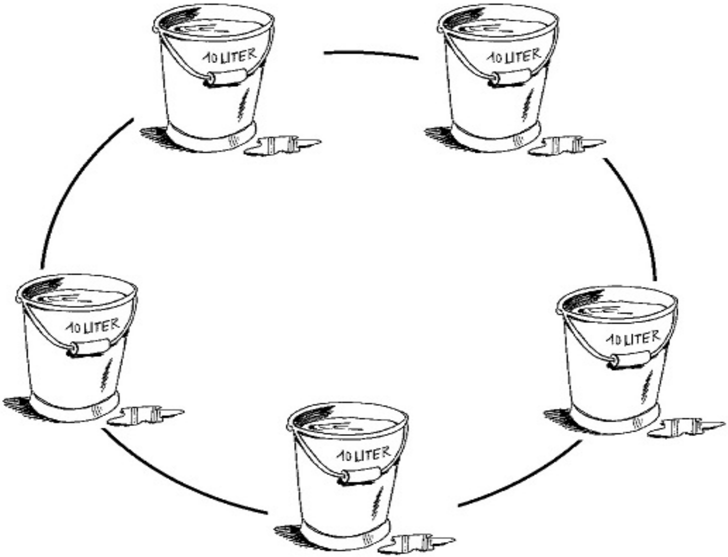
Cinderella versus the wicked Stepmother
or
How to avoid overflow

Marijke Bodlaender Cor Hurkens Vincent Kusters
Frank Staals Gerhard Woeginger Hans Zantema

TCS, Amsterdam, September 27, 2012

The game



- Five buckets in a circle of capacity b each

- Five buckets in a circle of capacity *b* each
- Initially all empty

- Five buckets in a circle of capacity *b* each
- Initially all empty
- Two players

- Five buckets in a circle of capacity b each
- Initially all empty
- Two players
- S(tepmother): tries to cause overflow by divide one liter of water over the buckets

- Five buckets in a circle of capacity b each
- Initially all empty
- Two players
- S(tepmother): tries to cause overflow by divide one liter of water over the buckets
- C(inderella): tries to avoid overflow by emptying two neighboring buckets

- Five buckets in a circle of capacity *b* each
- Initially all empty
- Two players
- S(tepmother): tries to cause overflow by divide one liter of water over the buckets
- C(inderella): tries to avoid overflow by emptying two neighboring buckets

For *b* small: S will win

- Five buckets in a circle of capacity *b* each
- Initially all empty
- Two players
- S(tepmother): tries to cause overflow by divide one liter of water over the buckets
- C(inderella): tries to avoid overflow by emptying two neighboring buckets

For *b* small: S will win

For *b* large: C will win

- Five buckets in a circle of capacity b each
- Initially all empty
- Two players
- S(tepmother): tries to cause overflow by divide one liter of water over the buckets
- C(inderella): tries to avoid overflow by emptying two neighboring buckets

For b small: S will win

For b large: C will win

Question: where is the border, that is, what is the smallest b for which C can avoid overflow?

Remarks

Remarks

- Non-trivial two-player game of hybrid nature: C behavior is discrete, S behavior is continuous

Remarks

- Non-trivial two-player game of hybrid nature: C behavior is discrete, S behavior is continuous
- Right answer: $b = 2$
  This was a problem in the shortlist of IMO (International Mathematics Olympiad), 2009

Remarks

- Non-trivial two-player game of hybrid nature: C behavior is discrete, S behavior is continuous
- Right answer: $b = 2$
  This was a problem in the shortlist of IMO (International Mathematics Olympiad), 2009
- We will generalize the problem to n buckets in a circle, and emptying c consecutive buckets

Remarks

- Non-trivial two-player game of hybrid nature: C behavior is discrete, S behavior is continuous
- Right answer: $b = 2$
  This was a problem in the shortlist of IMO (International Mathematics Olympiad), 2009
- We will generalize the problem to n buckets in a circle, and emptying c consecutive buckets
- Surprise: greedy strategy for C (always empty the two neighboring buckets with the highest amount of water) does *NOT* work for $n = 5$, $c = 2$, as we will show now

Start by several rounds in which S divides the water evenly over the buckets:

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{array}{ccc} 0 & & 1/5 \\ 0 & & 1/5 \\ 0 & \rightarrow_S & 1/5 \\ 0 & & 1/5 \\ 0 & & 1/5 \end{array}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{matrix} 0 & & 1/5 & & 1/5 \\ 0 & & 1/5 & & 1/5 \\ 0 & \rightarrow_S & 1/5 & \rightarrow_C & 1/5 \\ 0 & & 1/5 & & 0 \\ 0 & & 1/5 & & 0 \end{matrix}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{array}{ccccccc} 0 & & 1/5 & & 1/5 & & 8/25 \\ 0 & & 1/5 & & 1/5 & & 8/25 \\ 0 & \rightarrow_S & 1/5 & \rightarrow_C & 1/5 & \rightarrow_S & 8/25 \\ 0 & & 1/5 & & 0 & & 8/25 \\ 0 & & 1/5 & & 0 & & 8/25 \end{array}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{matrix} 0 & & 1/5 & & 1/5 & & 8/25 & & 1/2-\epsilon & & 1/2-\epsilon \\ 0 & & 1/5 & & 1/5 & & 8/25 & & 1/2-\epsilon & & 1/2-\epsilon \\ 0 & \rightarrow_S & 1/5 & \rightarrow_C & 1/5 & \rightarrow_S & 8/25 & \rightarrow^+ & 1/2-\epsilon & \rightarrow_C & 1/2-\epsilon \\ 0 & & 1/5 & & 0 & & 8/25 & & 1/2-\epsilon & & 0 \\ 0 & & 1/5 & & 0 & & 8/25 & & 1/2-\epsilon & & 0 \end{matrix}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{matrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{matrix} \to_S \begin{matrix} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{matrix} \to_C \begin{matrix} 1/5 \\ 1/5 \\ 1/5 \\ 0 \\ 0 \end{matrix} \to_S \begin{matrix} 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \end{matrix} \to^+ \begin{matrix} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \end{matrix} \to_C \begin{matrix} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 0 \\ 0 \end{matrix}$$

$$\to_S \begin{matrix} 1/2+\epsilon \\ 3/4-\epsilon' \\ 3/4-\epsilon' \\ 1/2+\epsilon \\ 0 \end{matrix}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array} \rightarrow_S \begin{array}{c} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{array} \rightarrow_C \begin{array}{c} 1/5 \\ 1/5 \\ 1/5 \\ 0 \\ 0 \end{array} \rightarrow_S \begin{array}{c} 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \end{array} \rightarrow^+ \begin{array}{c} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \end{array} \rightarrow_C \begin{array}{c} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 0 \\ 0 \end{array}$$

$$\rightarrow_S \begin{array}{c} 1/2+\epsilon \\ 3/4-\epsilon' \\ 3/4-\epsilon' \\ 1/2+\epsilon \\ 0 \end{array} \begin{array}{c} \text{greedy} \\ \rightarrow_C \\ \textit{wrong} \end{array} \begin{array}{c} 1/2+\epsilon \\ 0 \\ 0 \\ 1/2+\epsilon \\ 0 \end{array}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$
\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{array}
\to_S
\begin{array}{c} 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \\ 1/5 \end{array}
\to_C
\begin{array}{c} 1/5 \\ 1/5 \\ 1/5 \\ 0 \\ 0 \end{array}
\to_S
\begin{array}{c} 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \\ 8/25 \end{array}
\to^+
\begin{array}{c} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \end{array}
\to_C
\begin{array}{c} 1/2-\epsilon \\ 1/2-\epsilon \\ 1/2-\epsilon \\ 0 \\ 0 \end{array}
$$

$$
\to_S
\begin{array}{c} 1/2+\epsilon \\ 3/4-\epsilon' \\ 3/4-\epsilon' \\ 1/2+\epsilon \\ 0 \end{array}
\begin{array}{c} \text{greedy} \\ \to_C \\ \textit{wrong} \end{array}
\begin{array}{c} 1/2+\epsilon \\ 0 \\ 0 \\ 1/2+\epsilon \\ 0 \end{array}
\to_S
\begin{array}{c} 1+\epsilon \\ 0 \\ 0 \\ 1+\epsilon \\ 0 \end{array}
$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{array}{ccccccccccc} 0 & & 1/5 & & 1/5 & & 8/25 & & 1/2-\epsilon & & 1/2-\epsilon \\ 0 & & 1/5 & & 1/5 & & 8/25 & & 1/2-\epsilon & & 1/2-\epsilon \\ 0 & \rightarrow_S & 1/5 & \rightarrow_C & 1/5 & \rightarrow_S & 8/25 & \rightarrow^+ & 1/2-\epsilon & \rightarrow_C & 1/2-\epsilon \\ 0 & & 1/5 & & 0 & & 8/25 & & 1/2-\epsilon & & 0 \\ 0 & & 1/5 & & 0 & & 8/25 & & 1/2-\epsilon & & 0 \end{array}$$

$$\begin{array}{ccccccccc} & 1/2+\epsilon & & 1/2+\epsilon & & 1+\epsilon & & 1+\epsilon \\ & 3/4-\epsilon' & \text{greedy} & 0 & & 0 & & 0 \\ \rightarrow_S & 3/4-\epsilon' & \rightarrow_C & 0 & \rightarrow_S & 0 & \rightarrow_C & 0 \\ & 1/2+\epsilon & \textit{wrong} & 1/2+\epsilon & & 1+\epsilon & & 0 \\ & 0 & & 0 & & 0 & & 0 \end{array}$$

Start by several rounds in which S divides the water evenly over the buckets:

$$\begin{array}{c} 0\\0\\0\\0\\0 \end{array} \to_S \begin{array}{c} 1/5\\1/5\\1/5\\1/5\\1/5 \end{array} \to_C \begin{array}{c} 1/5\\1/5\\1/5\\0\\0 \end{array} \to_S \begin{array}{c} 8/25\\8/25\\8/25\\8/25\\8/25 \end{array} \to^+ \begin{array}{c} 1/2-\epsilon\\1/2-\epsilon\\1/2-\epsilon\\1/2-\epsilon\\1/2-\epsilon \end{array} \to_C \begin{array}{c} 1/2-\epsilon\\1/2-\epsilon\\1/2-\epsilon\\0\\0 \end{array}$$

$$\to_S \begin{array}{c} 1/2+\epsilon\\3/4-\epsilon'\\3/4-\epsilon'\\1/2+\epsilon\\0 \end{array} \begin{array}{c} \text{greedy}\\ \to_C \\ \textit{wrong} \end{array} \begin{array}{c} 1/2+\epsilon\\0\\0\\1/2+\epsilon\\0 \end{array} \to_S \begin{array}{c} 1+\epsilon\\0\\0\\1+\epsilon\\0 \end{array} \to_C \begin{array}{c} 1+\epsilon\\0\\0\\0\\0 \end{array} \to_S \begin{array}{c} 2+\epsilon\\0\\0\\0\\0 \end{array}$$

Overflow!

If $b < 2$ then S can always force overflow, independent of strategy of C:

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccc} 0 & & 1/2-\epsilon \\ 0 & & 1/2-\epsilon \\ 0 & \to_S \cdots \to_C & 1/2-\epsilon \\ 0 & & 0 \\ 0 & & 0 \end{array}$$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccccc} 0 & & 1/2-\epsilon & & 1-\epsilon \\ 0 & & 1/2-\epsilon & & 1/2-\epsilon \\ 0 & \rightarrow_S \cdots \rightarrow_C & 1/2-\epsilon & \rightarrow_S & 1-\epsilon \\ 0 & & 0 & & 0 \\ 0 & & 0 & & 0 \end{array}$$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{matrix} 0 & & 1/2-\epsilon & & 1-\epsilon & & 1-\epsilon \\ 0 & & 1/2-\epsilon & & 1/2-\epsilon & & 0 \\ 0 & \rightarrow_S \cdots \rightarrow_C & 1/2-\epsilon & \rightarrow_S & 1-\epsilon & \rightarrow_C & 0 \\ 0 & & 0 & & 0 & & 0 \\ 0 & & 0 & & 0 & & 0 \end{matrix}$$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccccccccc}
0 & & 1/2-\epsilon & & 1-\epsilon & & 1-\epsilon & & 2-\epsilon \\
0 & & 1/2-\epsilon & & 1/2-\epsilon & & 0 & & 0 \\
0 & \to_S \cdots \to_C & 1/2-\epsilon & \to_S & 1-\epsilon & \to_C & 0 & \to_S & 0 \\
0 & & 0 & & 0 & & 0 & & 0 \\
0 & & 0 & & 0 & & 0 & & 0
\end{array}$$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccccccccc}
0 & & 1/2-\epsilon & & 1-\epsilon & & 1-\epsilon & & 2-\epsilon \\
0 & & 1/2-\epsilon & & 1/2-\epsilon & & 0 & & 0 \\
0 & \rightarrow_S \cdots \rightarrow_C & 1/2-\epsilon & \rightarrow_S & 1-\epsilon & \rightarrow_C & 0 & \rightarrow_S & 0 \\
0 & & 0 & & 0 & & 0 & & 0 \\
0 & & 0 & & 0 & & 0 & & 0
\end{array}$$

is overflow since $2 - \epsilon > b$

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccccccccc} 0 & & 1/2-\epsilon & & 1-\epsilon & & 1-\epsilon & & 2-\epsilon \\ 0 & & 1/2-\epsilon & & 1/2-\epsilon & & 0 & & 0 \\ 0 & \rightarrow_S \cdots \rightarrow_C & 1/2-\epsilon & \rightarrow_S & 1-\epsilon & \rightarrow_C & 0 & \rightarrow_S & 0 \\ 0 & & 0 & & 0 & & 0 & & 0 \\ 0 & & 0 & & 0 & & 0 & & 0 \end{array}$$

is overflow since $2 - \epsilon > b$

So 2 is a *lower bound* for the smallest value b for which C can avoid overflow

If $b < 2$ then S can always force overflow, independent of strategy of C:

Choose $\epsilon > 0$ such that $\epsilon < 2 - b$

$$\begin{array}{ccccccccc} 0 & & 1/2-\epsilon & & 1-\epsilon & & 1-\epsilon & & 2-\epsilon \\ 0 & & 1/2-\epsilon & & 1/2-\epsilon & & 0 & & 0 \\ 0 & \to_S \cdots \to_C & 1/2-\epsilon & \to_S & 1-\epsilon & \to_C & 0 & \to_S & 0 \\ 0 & & 0 & & 0 & & 0 & & 0 \\ 0 & & 0 & & 0 & & 0 & & 0 \end{array}$$

is overflow since $2 - \epsilon > b$

So 2 is a *lower bound* for the smallest value b for which C can avoid overflow

More general:
Such a lower bound is found by fixing a strategy for S and show that this forces overflow for all (finitely many) choices of C steps

For finding the exact value we have to prove that the same value is also an *upper bound*, that is, there is a strategy for C for which for every strategy of S overflow is avoided

For finding the exact value we have to prove that the same value is also an *upper bound*, that is, there is a strategy for C for which for every strategy of S overflow is avoided

No finite analysis any more: S can divide one liter over the buckets in infinitely many ways

For finding the exact value we have to prove that the same value is also an *upper bound*, that is, there is a strategy for C for which for every strategy of S overflow is avoided

No finite analysis any more: S can divide one liter over the buckets in infinitely many ways

Approach for finding upper bound b:

For finding the exact value we have to prove that the same value is also an *upper bound*, that is, there is a strategy for C for which for every strategy of S overflow is avoided

No finite analysis any more: S can divide one liter over the buckets in infinitely many ways

Approach for finding upper bound b:

Find *invariant* *I* such that after every S step executed from a state satisfying *I*, C can do a step such that again *I* holds

For finding the exact value we have to prove that the same value is also an *upper bound*, that is, there is a strategy for C for which for every strategy of S overflow is avoided

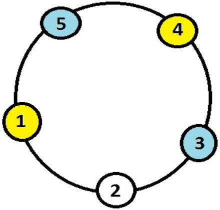
No finite analysis any more: S can divide one liter over the buckets in infinitely many ways

Approach for finding upper bound b:

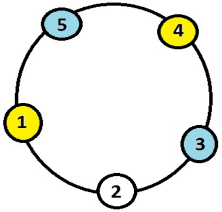
Find *invariant* I such that after every S step executed from a state satisfying I, C can do a step such that again I holds

If $I \Longrightarrow$ all values are $\leq b - 1$, then b is an upper bound

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty

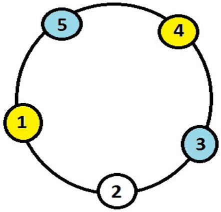


For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

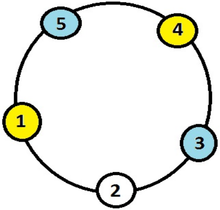
For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



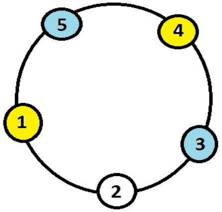
Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

After the S step: b_i = the contents of i

$$b_1 + b_3 + b_4 + b_5 < 2$$

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

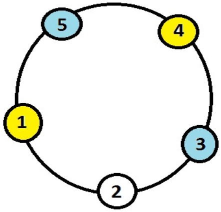
Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

After the S step: b_i = the contents of i

$$b_1 + b_3 + b_4 + b_5 < 2$$

Hence $b_1 + b_4 < 1$ or $b_3 + b_5 < 1$

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

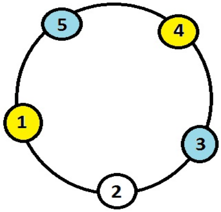
After the S step: b_i = the contents of i

$$b_1 + b_3 + b_4 + b_5 < 2$$

Hence $b_1 + b_4 < 1$ or $b_3 + b_5 < 1$

By symmetry we may assume $b_1 + b_4 < 1$

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

After the S step: b_i = the contents of i

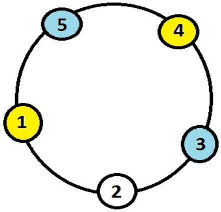
$$b_1 + b_3 + b_4 + b_5 < 2$$

Hence $b_1 + b_4 < 1$ or $b_3 + b_5 < 1$

By symmetry we may assume $b_1 + b_4 < 1$

Now C makes buckets 2 and 3 empty, so $4, 5, 1$ remain

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

After the S step: b_i = the contents of i

$$b_1 + b_3 + b_4 + b_5 < 2$$

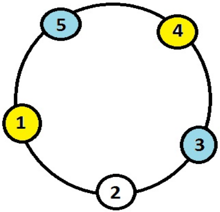
Hence $b_1 + b_4 < 1$ or $b_3 + b_5 < 1$

By symmetry we may assume $b_1 + b_4 < 1$

Now C makes buckets 2 and 3 empty, so $4, 5, 1$ remain

By $b_1 + b_4 < 1$ and $b_5 \leq 1$, the invariant I holds again

For our case $n = 5$, $c = 2$ after every C step two buckets have been made empty



Let these buckets be numbered 4 and 5, and a_i = contents of bucket i

Invariant I: $a_1 + a_3 < 1 \wedge a_2 \leq 1$

After the S step: b_i = the contents of i

$$b_1 + b_3 + b_4 + b_5 < 2$$

Hence $b_1 + b_4 < 1$ or $b_3 + b_5 < 1$

By symmetry we may assume $b_1 + b_4 < 1$

Now C makes buckets 2 and 3 empty, so $4, 5, 1$ remain

By $b_1 + b_4 < 1$ and $b_5 \leq 1$, the invariant I holds again

$I \Rightarrow$ all values are $\leq 2 - 1$, so $b = 2$ is an upper bound

Since we proved that $b = 2$ is both an upper bound and a lower bound, we conclude that $b = 2$ is the lowest value for which C can avoid overflow

Since we proved that $b = 2$ is both an upper bound and a lower bound, we conclude that $b = 2$ is the lowest value for which C can avoid overflow

How about other values of n, c?

Since we proved that $b = 2$ is both an upper bound and a lower bound, we conclude that $b = 2$ is the lowest value for which C can avoid overflow

How about other values of n, c?

As C tries to keep the contents of each of the buckets less than $b - 1$, we define $F(n, c) = b - 1$, for $1 \leq c < n$, and b is the lowest value for which C can avoid overflow in the game with n buckets in which at every C step c consecutive buckets are made empty

Since we proved that $b = 2$ is both an upper bound and a lower bound, we conclude that $b = 2$ is the lowest value for which C can avoid overflow

How about other values of n, c?

As C tries to keep the contents of each of the buckets less than $b - 1$, we define $F(n, c) = b - 1$, for $1 \leq c < n$, and b is the lowest value for which C can avoid overflow in the game with n buckets in which at every C step c consecutive buckets are made empty

So we proved $F(5, 2) = 1$

Results for $F(n, c)$ for $n \leq 12$

	c:	1	2	3	4	5	6	7	8	9	10	11
n:	2	1	–	–	–	–	–	–	–	–	–	–
	3	3/2	1/2	–	–	–	–	–	–	–	–	–
	4	11/6	1	1/3	–	–	–	–	–	–	–	–
	5	25/12	1	5/9	1/4	–	–	–	–	–	–	–
	6	137/60	3/2	1	1/2	1/5	–	–	–	–	–	–
	7	49/20	3/2	1	17/30	7/20	1/6	–	–	–	–	–
	8	363/140	11/6	1	1	1/2	1/3	1/7	–	–	–	–
	9	761/280	11/6	3/2	1	299/525	1/2	9/35	1/8	–	–	–
	10	7129/2520	25/12	3/2	1	1	5/9	69/196	1/4	1/9	–	–
	11	7381/2520	25/12	3/2	1	1	77/135	1/2	1/3	11/54	1/10	–
	12	83711/27720	137/60	11/6	3/2	1	1	5/9	1/2	1/3	1/5	1/11

Results for $F(n, c)$ for $n \leq 12$

	c:	1	2	3	4	5	6	7	8	9	10	11
n:	2	1	–	–	–	–	–	–	–	–	–	–
	3	3/2	1/2	–	–	–	–	–	–	–	–	–
	4	11/6	1	1/3	–	–	–	–	–	–	–	–
	5	25/12	1	5/9	1/4	–	–	–	–	–	–	–
	6	137/60	3/2	1	1/2	1/5	–	–	–	–	–	–
	7	49/20	3/2	1	17/30	7/20	1/6	–	–	–	–	–
	8	363/140	11/6	1	1	1/2	1/3	1/7	–	–	–	–
	9	761/280	11/6	3/2	1	299/525	1/2	9/35	1/8	–	–	–
	10	7129/2520	25/12	3/2	1	1	5/9	69/196	1/4	1/9	–	–
	11	7381/2520	25/12	3/2	1	1	77/135	1/2	1/3	11/54	1/10	–
	12	83711/27720	137/60	11/6	3/2	1	1	5/9	1/2	1/3	1/5	1/11

Hardest result: $F(n, c) = 1$ for $2 \leq n/c < 3$

How did we find these results?

How did we find these results?

Essentially the same as for $F(5,2)$:

How did we find these results?

Essentially the same as for $F(5, 2)$:

- Lower bound: define a strategy for S and show how for every value below the bound S can force overflow in finitely many steps, doing finite case analysis on C behavior

How did we find these results?

Essentially the same as for $F(5,2)$:

- Lower bound: define a strategy for S and show how for every value below the bound S can force overflow in finitely many steps, doing finite case analysis on C behavior
- Upper bound: define an invariant implying the bound on bucket values, and show that C can keep this invariant after her move for every possible S step

How did we find these results?

Essentially the same as for $F(5, 2)$:

- Lower bound: define a strategy for S and show how for every value below the bound S can force overflow in finitely many steps, doing finite case analysis on C behavior
- Upper bound: define an invariant implying the bound on bucket values, and show that C can keep this invariant after her move for every possible S step

combined with some general lemmas:

How did we find these results?

Essentially the same as for $F(5,2)$:

- Lower bound: define a strategy for S and show how for every value below the bound S can force overflow in finitely many steps, doing finite case analysis on C behavior
- Upper bound: define an invariant implying the bound on bucket values, and show that C can keep this invariant after her move for every possible S step

combined with some general lemmas:

Lemma

$F(\lambda n, \lambda c) = F(n, c)$ *for all integers* $\lambda \geq 1$.

How did we find these results?

Essentially the same as for $F(5,2)$:

- Lower bound: define a strategy for S and show how for every value below the bound S can force overflow in finitely many steps, doing finite case analysis on C behavior
- Upper bound: define an invariant implying the bound on bucket values, and show that C can keep this invariant after her move for every possible S step

combined with some general lemmas:

Lemma

$F(\lambda n, \lambda c) = F(n, c)$ for all integers $\lambda \geq 1$.

Lemma

If $n_1/c_1 \leq n_2/c_2$ then $F(n_1, c_1) \leq F(n_2, c_2)$

Typical S strategy for finding lower bound:

Typical S strategy for finding lower bound:

First divide water evenly during several rounds until some limit value has been nearly reached

Typical S strategy for finding lower bound:

First divide water evenly during several rounds until some limit value has been nearly reached

Next a limited number of rounds is played, in which typically S divides the water evenly among some subset of buckets

Typical S strategy for finding lower bound:

First divide water evenly during several rounds until some limit value has been nearly reached

Next a limited number of rounds is played, in which typically S divides the water evenly among some subset of buckets

For instance, for proving $F(9, 5) \geq 299/525$:

Typical S strategy for finding lower bound:

First divide water evenly during several rounds until some limit value has been nearly reached

Next a limited number of rounds is played, in which typically S divides the water evenly among some subset of buckets

For instance, for proving $F(9,5) \geq 299/525$:

The first phase of the balancing Stepmother ends with four consecutive buckets (say buckets 1, 2, 3, 4) having contents very close to $1/5$. The second phase goes through three further rounds.
In the first of these rounds, the Stepmother uses set $T_1 = \{1,2,3,4,6,7,8\}$ with all buckets except 5 and 9. The Stepmother brings every bucket in T_1 to contents $9/35$. Cinderella leaves a set of four buckets, at least three of which are in T_1. These three buckets are either adjacent (say 2, 3, 4 in this first case) or separated by a single empty bucket (say 3, 4, 6 in the second case).
In the second round the Stepmother selects the set T_2 to contain five buckets; in the first case she uses $T_2 = \{2,3,4,7,8\}$ and in the second case $T_2 = \{3,4,6,7,8\}$. The Stepmother brings every bucket in T_2 to contents $62/175$. Cinderella leaves a set of four buckets, at least two of which are in T_2. We rename the buckets so that 1 and $b \in \{2,3,4\}$ keep their contents $62/175$.

In the third round the Stepmother uses $T_3 = \{1, b, 6\}$, and fills these three buckets up to level $299/525$. Cinderella

must leave at least one such bucket with contents $299/525$ at the end of the round.

Typical invariant for concluding upper bound:
conjunction of bounds on subsets of k elements, for various values of k

Typical invariant for concluding upper bound:
conjunction of bounds on subsets of k elements, for various values of k

For instance, for proving $F(9,5) \leq 299/525$ the invariant is the conjunction of

Typical invariant for concluding upper bound: conjunction of bounds on subsets of k elements, for various values of k

For instance, for proving $F(9,5) \leq 299/525$ the invariant is the conjunction of

$$
\begin{array}{rclcll}
x_i & < & 299/525 & \approx & 0.569 & \text{for } 1 \leq i \leq 9 \\
x(S) & < & 124/175 & \approx & 0.708 & \text{for all } S \text{ with } |S| = 2 \\
x(S) & < & 27/35 & \approx & 0.771 & \text{for all } S \text{ with } |S| = 3 \\
x(S) & < & 4/5 & = & 0.800 & \text{for all } S \text{ with } |S| = 4
\end{array}
$$

Typical invariant for concluding upper bound: conjunction of bounds on subsets of k elements, for various values of k

For instance, for proving $F(9,5) \leq 299/525$ the invariant is the conjunction of

$$\begin{array}{rclcll} x_i & < & 299/525 & \approx & 0.569 & \text{for } 1 \leq i \leq 9 \\ x(S) & < & 124/175 & \approx & 0.708 & \text{for all } S \text{ with } |S| = 2 \\ x(S) & < & 27/35 & \approx & 0.771 & \text{for all } S \text{ with } |S| = 3 \\ x(S) & < & 4/5 & = & 0.800 & \text{for all } S \text{ with } |S| = 4 \end{array}$$

These choices are quite subtle: minor variants of this formula are not invariant any more

How did we find these proofs?

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

Upper bound by invariant: we developed machinery for checking invariants:

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

Upper bound by invariant: we developed machinery for checking invariants:

- input: n, c and desired invariant

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

Upper bound by invariant: we developed machinery for checking invariants:

- input: n, c and desired invariant
- output: either the observation that C can indeed maintain this invariant, or a state and an S-step for which C can not do a step maintaining the invariant

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

Upper bound by invariant: we developed machinery for checking invariants:

- input: n, c and desired invariant
- output: either the observation that C can indeed maintain this invariant, or a state and an S-step for which C can not do a step maintaining the invariant

This machinery was crucial for being able to find the right invariants

How did we find these proofs?

Lower bound by choosing strategy for S: by hand

Upper bound by invariant: we developed machinery for checking invariants:

- input: n, c and desired invariant
- output: either the observation that C can indeed maintain this invariant, or a state and an S-step for which C can not do a step maintaining the invariant

This machinery was crucial for being able to find the right invariants

Finally, for all cases with $n \leq 10$ we could give the invariance proofs by hand

Main ingredient of the machinery:

Main ingredient of the machinery:

SMT (satisfiability modulo theories) using the tool YICES

Main ingredient of the machinery:

SMT (satisfiability modulo theories) using the tool YICES

Feasibility in linear programming: check whether a *conjunction* of linear inequalities admits a solution

Main ingredient of the machinery:

SMT (satisfiability modulo theories) using the tool YICES

Feasibility in linear programming: check whether a *conjunction* of linear inequalities admits a solution

SMT: check whether a *propositional formula* in linear inequalities admits a solution

Main ingredient of the machinery:

SMT (satisfiability modulo theories) using the tool YICES

Feasibility in linear programming: check whether a *conjunction* of linear inequalities admits a solution

SMT: check whether a *propositional formula* in linear inequalities admits a solution

I is an invariant if the formula

$$I \;\wedge\; \left(\sum_i y_i = 1 + \sum_i x_i\right) \;\wedge\; \left(\bigwedge_i y_i \geq x_i\right) \;\wedge\; \neg \bigvee_j I_j$$

is unsatisfiable

Main ingredient of the machinery:

SMT (satisfiability modulo theories) using the tool YICES

Feasibility in linear programming: check whether a *conjunction* of linear inequalities admits a solution

SMT: check whether a *propositional formula* in linear inequalities admits a solution

I is an invariant if the formula

$$I \;\wedge\; \left(\sum_i y_i = 1 + \sum_i x_i\right) \;\wedge\; \left(\bigwedge_i y_i \geq x_i\right) \;\wedge\; \neg \bigvee_j I_j$$

is unsatisfiable

Here the formula I_j is obtained from I by replacing x_i by y_i for $n - c$ consecutive i's, and replacing the other c by 0, for all $j = 1, \ldots, n$ possibilities to do so

Concluding remarks

Concluding remarks

- We investigated a two-player game

Concluding remarks

- We investigated a two-player game
  - parameterized by integers $0 < c < n$ and real $b > 0$

Concluding remarks

- We investigated a two-player game
  - parameterized by integers $0 < c < n$ and real $b > 0$
  - hybrid: one player (C) has discrete behavior, the other (S) continuous

Concluding remarks

- We investigated a two-player game
  - parameterized by integers $0 < c < n$ and real $b > 0$
  - hybrid: one player (C) has discrete behavior, the other (S) continuous
  - S wins in finitely many steps, C wins by playing forever

Concluding remarks

- We investigated a two-player game
  - parameterized by integers $0 < c < n$ and real $b > 0$
  - hybrid: one player (C) has discrete behavior, the other (S) continuous
  - S wins in finitely many steps, C wins by playing forever
  - find minimal value of b for which C can win

Concluding remarks

- We investigated a two-player game
  - parameterized by integers $0 < c < n$ and real $b > 0$
  - hybrid: one player (C) has discrete behavior, the other (S) continuous
  - S wins in finitely many steps, C wins by playing forever
  - find minimal value of b for which C can win
- Definition of game very simple, solutions very hard, already for very low parameters obvious greedy strategy is not optimal

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by
  - Find a strategy for S such that for every $a < b$ the value a can be reached

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by
  - Find a strategy for S such that for every $a < b$ the value a can be reached
  - Find an invariant by which C can force that all values stay below b

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by
  - Find a strategy for S such that for every $a < b$ the value a can be reached
  - Find an invariant by which C can force that all values stay below b
- For the latter we were strongly guided by computer support by means of SMT solving

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by
  - Find a strategy for S such that for every $a < b$ the value a can be reached
  - Find an invariant by which C can force that all values stay below b
- For the latter we were strongly guided by computer support by means of SMT solving
- Elegant interplay between human proof search and computer support

Concluding remarks, continued

- We fully investigated all cases with $n \leq 12$, resulting in fancy quotients
- All our results were found by
  - Find a strategy for S such that for every $a < b$ the value a can be reached
  - Find an invariant by which C can force that all values stay below b
- For the latter we were strongly guided by computer support by means of SMT solving
- Elegant interplay between human proof search and computer support
- It is open whether this approach works for all $0 < c < n$, and whether resulting values are always rational