Antirandomizing the Wrong Game

Benjamin Doerr

Mathematisches Seminar II, Christian–Albrechts–Universität zu Kiel,
Ludewig–Meyn–Str. 4, D–24098 Kiel, Germany,
bed@numerik.uni-kiel.de,
WWW home page: http://www.numerik.uni-kiel.de/~bed/

Abstract. We study a variant of the tenure game introduced by J. Spencer (Theor. Comput. Sci. 131 (1994), 415–429). In this version, chips are not removed from the game, but moved down to the lowest level instead. Though the rules of both versions differ only slightly, it seems impossible to convert an upper bound strategy into a lower bound one using the antirandomization approach of Spencer (which was very effective for the original game and several others).
For the upper bound we give a potential function argument (both randomized and derandomized). We manage to prove a nearly matching lower bound using a strategy that can be interpreted as an antirandomization of Spencer's original game.
Key words: Games, randomization, derandomization.

1 Introduction

Since the increasing interest in on-line problems at the latest, game theory has gained attraction in theoretical computer science. In this paper we work on so-called Pusher-Chooser games. These are two player perfect information games where each round the player called 'Pusher' splits the position into two alternatives and 'Chooser' selects one thereof. Hence the theme of these games is on-line balancing: Pusher has to find a balanced split (in the sense that neither alternative is too favorable to Chooser), whereas Chooser tries to detect and exploit such imbalances. Examples of such games are vector balancing games ([2, 3, 6, 7, 9, 10]) and liar games. Concerning the latter, we refer to the survey [8] its extensive bibliography of 120 references. Internet routing problems gave rise to the related "guessing secrets" problem, that attracted attention recently ([1, 4, 5]).

In his marvelous paper "Randomization, Derandomization and Antirandomization: Three Games", Spencer [11] shows a generic method to convert a random strategy for Chooser in such a game into a deterministic algorithm. Moreover, he also provides a method coined 'antirandomization' that produces a matching lower bound constructively, i.e., including a strategy for Pusher.

The game Spencer demonstrated these methods most easily is the *Tenure game*. We cite the rules from [11]:

> The tenure game is a perfect information game between two players, Paul — chairman of the department — and Carole — dean of the school. An initial position is given in which various faculty are at various pre-tenured positions. Paul will win if some faculty member receives tenure — Carole wins if no faculty member receives tenure. Each year Chair Paul creates a promotion list L of the faculty and gives it to Dean Carole who has two options: (1) Carole may promote all faculty on list L one rung and simultaneously fire all other faculty. (2) Carole may promote all faculty *not* on list L one rung and simultaneously fire all faculty on list L.

In this paper we study a slight variant of the Tenure game introduced by Spencer. We will assume that not-promoted faculty is not fired, but downgraded to the first rung instead. There are two reasons to investigate this game. In [6] we showed that good strategies for this game yield good strategies for the on-line vector balancing problem with aging, i.e., where decisions in the past become less important compared to the actual one. For reasons of space we just refer to the paper for this aspect.

Our main motivation is that this variant — though similar to the original game and clearly a Pusher-Chooser game — seemingly does not admit antirandomization. Since the antirandomization method is a very powerful way to convert a randomized or derandomized strategy for Chooser into a matching strategy for Pusher, it is particularly interesting to investigate what happens if it cannot be applied. This work suggests two answers: First, things can become quite complicated without antirandomization, second, sometimes using the antirandomization of a different game can help.

Let us state the rules precisely. Whether Carole can win or not of course depends on the number of rungs we have. To remove this parameter without losing information about the game, we assume to have infinitely many rungs and play an optimization version of the game: The highest rung reached by some faculty member is called the pay-off for Paul. Naturally, he tries to maximize this pay-off, whereas Carole tries to minimize it. To further simplify the setting we assume that all faculty is on the first rung at the beginning of the game. Exchanging people by innocent chips and baptizing this version 'European Tenure Game', we have:

> **Rules of the European Tenure Game:** The game is played with a fixed number d of chips which lie on levels numbered with the positive integers. At the start of the game, all d chips are on level one. The game is a two-player perfect information game. The first player, called 'Pusher', tries to get a chip to a possibly high level. The maximum level ever reached by a chip during the game is called pay-off to Pusher. Each round Pusher chooses a subset of chips he proposes to be promoted. If the second player ('Chooser') accepts, then these chips each move up one level, and the remaining chips are moved down to the first level. If Chooser rejects, the the remaining chips move up one level, and Chooser's choice is downgraded to level one. The game ends if pusher is satisfied with the position reached or a position reoccurs.

From the rules it is already clear that Pusher has some advantage in the European Tenure Game compared to Spencer's original game, which we shall call 'American Tenure Game'.

For the American Tenure Game, Spencer gave a complete solution even for arbitrary starting positions (which we do not regard in this paper). If the game is played with d chips, then the value of this game, i.e., the maximum level Pusher can reach, is $\lfloor \log d \rfloor + 1$, where $\log(\cdot)$ shall always denote the dyadic logarithm.

The European Tenure Game seems to be more difficult to analyze. Using straightforward reasoning, a bound of $\lfloor \log d \rfloor + \lfloor \log \log d \rfloor \leq v_d \leq \log d + \log \log d + 4$ for the value v_d of this game was shown in [6], where the game appeared first in a reduction of a vector balancing problem. In this paper we make some progress towards a full understanding of the game.

For the general case, we reduce the gap between lower and upper bound, so that there at most three possibilities for each d. For larger d, the gap reduces to at most two values, though we are able to determine a precise value for a set having positive lower density. To prove the lower bound we analyze a strategy that seems to be an antirandomization of the American Tenure Game strategy.

Theorem 1. *Let v_d denote the value of the European Tenure Game played with d chips. Then*

$$\lfloor \log d + \log \log d \rfloor \leq v_d \leq \lfloor \log d + \log \log d + 1.98 \rfloor$$

holds for all d. For d tending to infinity, these bounds improve to

$$\lfloor \log d + \log \log d + 1 + o(1) \rfloor \leq v_d \leq \lfloor \log d + \log \log d + 1.73 + o(1) \rfloor .$$

In particular, the set $S = \{d \in \mathbb{N} \,|\, v_d = \lfloor \log d + \log \log d + 1 \rfloor\}$ has lower density greater than $\frac{1}{5}$.[1]

2 Upper Bound: Chooser's Strategy

Let us assume $d \geq 3$ since smaller cases are trivial. We describe a position of the game by a function $P : \mathbb{N} \to \mathbb{N}_0$ such that $\sum_{i \in \mathbb{N}} P(i) = d$. Hence $P(i)$ denotes the number of chips on level i.

For the upper bound we are guided by [6]. Let $\lambda := \frac{2 \log d - 1}{\log d}$. Define a potential function v by $v(P) := \sum_{i \in \mathbb{N}} P(i) \lambda^{i-1}$ for all positions $P : \mathbb{N} \to \mathbb{N}_0$. We analyze the strategy for Chooser to choose that one of the alternatives which minimizes $v(P)$ for the resulting position. An easy induction shows that this ensures $v(P) \leq d \log d$ for all positions P occurring in the game, which in turn yields an upper bound for the value of the game.

[1] The lower density $\underline{d}(S)$ of a set $S \subseteq \mathbb{N}$ is $\underline{d}(S) := \liminf_{n \to \infty} \frac{1}{n} |\{s \in S \,|\, s \leq n\}|$. Roughly speaking, the last paragraph of the theorem states that we know the precise value of the game for more than a fifth of the values for d.

Lemma 1. *The value of the game played with d chips is at most*

$$\frac{\log(d\log d - d + 1)}{\log\left(2 - \frac{1}{\log d}\right)} + 1 \leq \log d + \log\,\log d + 1.73 + o(1).$$

Proof. Clearly we have $v(P) \leq d \log d$ for the starting position. Now let P be an arbitrary position of the game such that $v(P) \leq d\log d$. Denote by P_1, P_2 the two positions resulting from either accepting or rejecting Pusher's choice. Then

$$\begin{aligned}\min\{v(P_1), v(P_2)\} &\leq \tfrac{1}{2}(v(P_1) + v(P_2)) \\ &= \tfrac{1}{2}(d + \lambda v(P)) \\ &\leq \tfrac{1}{2}(d + \tfrac{2\log d - 1}{\log d} d \log d) \\ &= d\log d.\end{aligned}$$

Hence Chooser's strategy of minimizing $v(P)$ ensures that $v(P) \leq d\log d$ holds throughout the game.

Let P be any position such that $v(P) \leq d\log d$. Let l denote the level of the highest-ranking chip. Since the remaining $d - 1$ chips at least are on level one[2], we have

$$\lambda^{l-1} \leq d\log d - d + 1.$$

Hence

$$l \leq \log_\lambda(d\log d - d + 1) + 1 = \frac{\log(d\log d - d + 1)}{\log\left(2 - \frac{1}{\log d}\right)} + 1.$$

For $d \geq 3$, the latter term is strictly less than $\log d + \log\,\log d + 1.98$. For d tending to infinity, our bound becomes stronger and is optimal for quite a portion a values (as we will prove in the next section). We have

$$\frac{\log(d\log d - d + 1)}{\log\left(2 - \frac{1}{\log d}\right)} + 1 = \log d + \log\,\log d + 1 + \frac{1}{2\ln 2} + o(1).$$

For our purposes, the upper bound suffices.

Put $l = \log d$. Then $\log(2 - 1/l) = 1 + \log(l - 1/2) - \log l \geq 1 - \frac{1}{2}\frac{1}{(l-1/2)\ln 2}$, as the logarithm is concave. Thus

$$\begin{aligned}\frac{\log(d\log d - d + 1)}{\log\left(2 - \frac{1}{\log d}\right)} &\leq \frac{l + \log l}{\log(2 - 1/l)} \\ &\leq \frac{l + \log l}{1 - 1/(2(l - \frac{1}{2})\ln 2)} \\ &= l + \log l + \tfrac{1}{2\ln 2} + \tfrac{\log l}{2l\ln 2 - 1 - \ln 2} + \tfrac{1 + \ln 2}{2\ln 2(2l\ln 2 - 1 - \ln 2)}.\end{aligned}$$

□

[2] This seems to be a negligible advantage. For large d in fact it is, but for smaller values this is enough to reduce the upper bound from $l + \log l + 4$ to $l + \log l + 1.98$.

Above we gave a deterministic strategy. Assume now that Chooser plays randomly, i.e., he flips a fair coin to decide which of the two alternatives to take. Then a similar argument as above shows that the expected v–value is bounded by $d \log d$, no matter what strategy Pusher is playing. Since the American Tenure Game is a finite perfect information game, we deduce that Chooser actually has a strategy ensuring $v(P) \leq d \log d$ throughout the game. Moreover, the one we proposed first is just the derandomization of this randomized proof.

What makes this game interesting, is that the corresponding antirandomization does not work.

3 Lower Bound: Pusher's Strategy

The antirandomization paradigm of Spencer's paper [11] would advise Pusher to match Chooser's strategy this way: "Play each round such that the outcome of both alternatives to Chooser has the same potential $v(\cdot)$. Thus Chooser can never gain an advantage."

For several reasons this does not work. Firstly, there is not analogue to the splitting lemma in [11]. Thus in general it is not possible to split the position into nearly equally valued (with respect to v) alternatives. Secondly, the starting position does not have a potential of $d \log d$, but only of d. Thus equally valued splits, even if they existed, would not be enough. A third point is that the potential function does not yield a good lower bound: A high potential $v(P)$ does not guarantee that there is a chip on a high level. To some extent it does, but these bounds are not strong enough. The reason is that the potential function v values chips on a lower level slightly higher than Spencer's potential function for the American Tenure Game. During play this is justified by the fact that these chips gain from the advantage of not being fired, but just downgraded. At the end of the game, this advantage does not exist anymore.

For these reasons an antirandomization argument corresponding to our upper bound strategies seems not to work. What does work, however, is the antirandomization of the American Tenure Game. The prize for using a non-corresponding antirandomization is that the proofs are more complicated.

3.1 Strategies for the American Tenure Game

At this point let us shortly review Chooser's derandomized strategy in the American Tenure Game. Recall that we described a position of the game by a function $P : \mathbb{N} \to \mathbb{N}_0$ such that $\sum_{i \in \mathbb{N}} P(i) = d$. Chooser can follow the strategy to choose that one of the alternatives that minimizes $f(P) = \sum_{i \in \mathbb{N}} P(i) 2^i$ for the resulting position. An easy calculation shows that this way the f–value (f–potential) of the position cannot increase during play. Thus it never exceeds the initial value of $2d$, and it is clear that no chip can reach a higher level than $\log d + 1$.

The corresponding antirandomization yields this strategy for Pusher: He always chooses a split which maximizes the minimum f–potential among the two alternatives to Chooser. Thus he maximizes the f–potential of the resulting position regardless of Chooser's move. For the American Tenure Game it can be shown that playing this way the f–potential can never drop below $2^{\lfloor \log d \rfloor}$, which matches the upper bound. Crucial for this result is the so-called splitting lemma.

In the following, we analyze this same strategy of maximizing the f–potential for Pusher in the European Tenure Game. The fact that this is not an antirandomization of our strategy for Chooser makes the proofs somewhat harder, but nevertheless we end up with a near-tight bound.

3.2 First Phase

The case $d = 2$ is solved by a moment's thought, so let us assume $d \geq 3$. We assume first that $d = 2^l$ is a power of 2 and deal with the general case at the end of this section. We shortly review the result in [6], as we use this as first part of our strategy.

It is clear that Pusher can change a position P such that all $P(i)$ are even, to the position P' defined by $P'(1) = \frac{1}{2}d$ and $P'(i+1) = \frac{1}{2}P(i)$ for all $i \in \mathbb{N}$. All pusher has to do is to select half of the chips of each level. Then, regardless of Chooser's choice, he ends up with position P'. We call this procedure an 'easy split'.

From the starting position with $d = 2^l$ chips on the first level, Pusher can do l easy splits and reach a position P with $P(i) = 2^{l-i}$ for all $i = 1, \ldots, l$, with $P(l+1) = 1$ and $P(i) = 0$ for $i \geq l+2$. Doing so was part of the strategy in [6], and will be part of ours as well. The interesting point is how to continue from this position. In [6] we gave an explicit strategy moving one chip up to level $l + \lfloor \log l \rfloor$. Spencer (private communication) noted that the position P has an f–potential of $(l+2)d$. Thus a pay-off of $l + \lfloor \log (l+2) \rfloor$ can already be obtained in the American Tenure Game, which — as noted above — is less favorable for Pusher in the sense that he can get at most the same pay-off as in the European Tenure Game.

3.3 Second Phase

Guided by Spencer's observation, we now continue with a strategy that maximizes f. In the remainder of this paper, we will call $f(P)$ simply the potential of P omitting the f. Since in Phase 1 a greedy strategy of maximizing the function f was successful, one might be tempted to continue this. As each level has potential d (except level $l+1$, which has potential $2d$), it is not too difficult to split the levels into two parts having equal potential.[3] Thus the surviving part carries the whole potential (recall that moving up doubles the potential of a chip), and we gain a potential of 2 for each chip that is downgraded. We can continue this roughly $\log l$ times. If, while doing so, we partition the downgraded chips evenly,

[3] From the rules of the tenure games it is clear that it makes no difference whether Pusher proposes some set of chips or its complement. Therefore we may view any Pusher move simply as partition of the set of chips into two classes.

we can gain an extra potential of roughly $d \log l$. Since we needed roughly dl extra to prove our main result, we are not done yet.

The problem is that having played this way, we might end up with one chip on level $l + \log l$ holding most of the potential of the whole position. Hence Chooser will downgrade this chip in his next move, and all our clever gains are gone.

The solution is modesty. Of course, we cannot prevent the chip on level $l+1$ to move up to $l + \log l$ in $\log l - 1$ moves. Chooser can enforce this by simply downgrading that part of the chips that does not contain this highest-ranking one. Therefore, we partition the chips into classes having different potential: The one containing the highest-ranking chips has a that large potential, that we are immediately satisfied if it survives (ending with a potential of at least $2dl$). On the other hand, if the 'lower class' chips survive, we gain only little potential (an additional d), but end up with a flexible position (in particular having no too high-ranking chips, and allowing a similar step again). Here are the details:

We call the position

$$P_k : \mathbb{N} \to \mathbb{N}_0 ; i \mapsto \begin{cases} 2^{l-i} & \text{if } i < k \\ 1 & \text{if } i = k \\ 2^{l+1-i} & \text{if } k < i \leq l+1 \\ 0 & \text{otherwise.} \end{cases}$$

a *logarithmic ladder with gap at level* k. Further on, we define for all $0 \leq j < k$

$$Q_{k,j} : \mathbb{N} \to \mathbb{N}_0 ; i \mapsto \begin{cases} d(1-2^{-j}) & \text{if } i = 1 \\ 2^{l+1-i} & \text{if } j+2 \leq i \leq k \\ 1 & \text{if } i = k+1 \\ 2^{l+2-i} & \text{if } k+2 \leq i \leq l+2 \\ 0 & \text{otherwise.} \end{cases}$$

Lemma 2. *From a logarithmic ladder with gap P_k, Pusher can enforce for any $j < k$ one of the positions P_{l+1-j} and $Q_{k,j}$. In particular, he can advance from P_k to one of P_{k-1} and $Q_{k,l+2-k}$, if $k > l/2 + 1$.*

Proof. In position P_k, Pusher chooses those chips that have level at most j. If Chooser rejects, these $d(1-2^{-j})$ chips move down to level one, the remaining move up one level and position $Q_{k,j}$ is reached. Hence suppose that Chooser accepts. Then $d2^{-j} = 2^{l-j}$ chips move to level one, and the other chips move up one level. Now the number of chips on each level is a multiple of 2^{l-j}. Thus Pusher can play $l-j$ easy splits and reach position P_{l+1-j}. The second claim follows from the first by choosing $j = l+2-k$. □

Lemma 3. *For all $0 \leq j < k \leq l+1$, we have*

$$f(P_k) = d(2l - k + 1) + 2^k,$$
$$f(Q_{k,j}) = d(4l - 2(k+j) + 4) + 2^{k+1} - 2^{l-j+1}).$$

The proof is a simple calculation. Note that the levels of P_k below the gap (except the first one) each contribute d to the potential, whereas those above contribute $2d$.

From what we showed so far we already get a first lower bound:

Lemma 4. *For any $\lceil (l+1)/2 \rceil \leq s \leq l+1$, Pusher has a strategy enforcing one of the positions $Q_{k,l+2-k}$ for $k = s+1, \ldots, l$, and P_s. For $s = \lceil (l+1)/2 \rceil$, this strategy yields a potential of at least $1.5 d \log d$, and thus a lower bound for the value of the game of $\lfloor \log d + \log \log d + 0.58 \rfloor$.*

Proof. From the starting position with 2^l chips on level one, Pusher does l easy splits and reaches position $P_l = P_{l+1}$ (this is Phase 1). Once in Position P_i for some $l \geq i \geq s+1$, he applies Lemma 2 with $j = l+2-i$ and reaches $Q_{i,j}$ or P_{i-1}. This proves the statement concerning the possible positions. With $s = \lceil (l+1)/2 \rceil$, the bound for the value of the game follows directly from Lemma 3 and the discussion of the American Tenure Game. □

Since the positions $Q_{k,l+2-k}$ all have a potential of more than $2dl$, the lower bounds of Lemma 4 just depend on the potential of $P_{\lceil (l+1)/2 \rceil}$ of about $\frac{3}{2} dl$. We therefore continue Pusher's strategy on this position.

3.4 Third Phase

The reason why we could not continue applying Lemma 2 is that the gap k and the position j where Pusher splits the levels would meet. Splitting the levels above the gap leads to slightly more complicated positions having two gaps. For $0 < r < s \leq l+2$, we define

$$P_{r,s} : \mathbb{N} \to \mathbb{N}_0 ; i \mapsto \begin{cases} 2^{l-i} & \text{if } i < r \\ 1 & \text{if } i = r \\ 2^{l+1-i} & \text{if } r < i < s \\ 0 & \text{if } i = s \\ 2^{l+2-i} & \text{if } s < i \leq l+2 \\ 0 & \text{otherwise.} \end{cases}$$

We also need for $0 < r < j < s \leq l+2$

$$Q_{r,s,j} : \mathbb{N} \to \mathbb{N}_0 ; i \mapsto \begin{cases} d(1 - 2^{-j+1}) & \text{if } i = 1 \\ 1 & \text{if } i = r+1 \\ 2^{l+2-i} & \text{if } j+2 \leq i < s+1 \\ 0 & \text{if } i = s+1 \\ 2^{l+3-i} & \text{if } s+1 < i \leq l+3 \\ 0 & \text{otherwise.} \end{cases}$$

Again we compute their potentials:

Lemma 5.

$$f(P_{r,s}) = d(4l - r - 2s + 5) + 2^r$$
$$f(Q_{r,s,j}) = d(8l - 4s - 4j + 14) + 2^{r+1} - 2^{l-j+2}.$$

The following lemma shows that also logarithmic ladders with two gaps allow comprehensible strategies.

Lemma 6. *Let $0 < r < s \leq l+2$. For any j such that $r < j < s$, Pusher can advance position $P_{r,s}$ to one of $P_{l+2-j,l+r+2-j}$ and $Q_{r,s,j}$.*

Proof. Pusher chooses all chips on level at most j except the single chip on level r. If Chooser rejects, we are immediately in position $Q_{r,s,j}$. Otherwise, 2^{l-j+1} chips move to level one and Pusher's choice moves up one level. As all levels hold a multiple of 2^{l-j+1} chips, Pusher can play $l-j+1$ easy splits and reach position $P_{l+2-j,l+r+2-j}$. □

Using Lemma 6, we apply a modesty strategy again: By Lemma 4 (and one extra step if l is odd), we reach $P_{(l+1)/2,l+1}$ or $P_{(l+2)/2,l+2}$. Once in position $P_{x,2x}$ for some $x \in [\lfloor (x+7)/3 \rfloor, \lfloor (l+2)/2 \rfloor]$, Pusher slowly increases the potential through the position $P_{x-1,2x-1}$ to $P_{x-1,2(x-1)}$. The first step increases the potential by roughly $3d$, the second by $2d$. If Chooser tries to foil this strategy, he immediately ends up with a Q–position having a potential of roughly $2dl$. Apart from a few small cases, this leads to a potential greater than $2dl$.

Lemma 7. *In the European Tenure Game played with $d = 2^l$ chips, Pusher can reach one of the positions*

- $Q_{k,l+2-k}$ *for* $k = \lceil (l+1)/2 \rceil + 1, \dots, l$,
- $Q_{x,2x,l+3-x}$ *for* $x = \lfloor (l+7)/3 \rfloor, \dots, \lfloor (l+2)/2 \rfloor$,
- $Q_{x-1,2x-1,l+3-x}$ *for* $x = \lfloor (l+7)/3 \rfloor, \dots, \lfloor (l+3)/2 \rfloor$,
- $P_{\lfloor (l+4)/3 \rfloor, 2\lfloor (l+4)/3 \rfloor}$.

In consequence, Pusher can reach a potential of more than $2d(l-1)$.

Proof. Applying Lemma 4 with $s = \lceil (l+1)/2 \rceil$, Pusher can get one of the positions $Q_{k,l+2-k}$ for $k = \lceil (l+3)/2 \rceil, \dots, l$, or $P_{\lceil (l+1)/2 \rceil}$. Note that $P_{\lceil (l+1)/2 \rceil} = P_{\lceil (l+1)/2 \rceil, l+2}$.

If l is odd, we apply Lemma 6 with $j = \frac{l+1}{2} + 1$ and end up with either $Q_{(l+1)/2,l+2,(l+3)/2}$ (which is $Q_{x-1,2x-1,l+3-x}$ for $x = \lfloor (l+3)/2 \rfloor$) or $P_{(l+1)/2,l+1}$. If l is even, our actual position is $P_{(l+2)/2,l+2}$.

The rest is an easy induction: Assume that we are in position $P_{x,2x}$ for some $x = \lfloor (l+7)/3 \rfloor, \dots, \lfloor (l+2)/2 \rfloor$. Note that this implies $l \geq 4$. Applying Lemma 6 with $j = l+3-x$ on this position, we get $Q_{x,2x,l+3-x}$ or $P_{x-1,2x-1}$. Applying Lemma 6 on the latter with $j = l+3-x$ again, we reach position $Q_{x-1,2x-1,l+3-x}$ or $P_{x-1,2(x-1)}$. This proves the claim concerning the reachable positions.

For the potentials we look up in Lemma 3 and 5 and compute:

$$\begin{aligned} f(Q_{k,l+2-k}) &= 2dl + 3 \cdot 2^{k-1}, \\ f(Q_{x,2x,l+3-x}) &= 4d(l-x) + 2d + 3 \cdot 2^{x-1}, \\ f(Q_{x-1,2x-1,l+3-x}) &= 4d(l-x) + 6d + 2^{x-1}, \\ f(P_{\lfloor (l+4)/3 \rfloor, 2\lfloor (l+4)/3 \rfloor}) &= d(4l - 5\lfloor (l+4)/3 \rfloor + 5) + 2^{\lfloor (l+4)/3 \rfloor}. \end{aligned}$$

□

Note that all potentials above except the one of $Q_{x,2x,l+3-x}$ for $x = l/2 + 1$ and even $l \geq 4$ are at least $2dl$. We may remark that with some more effort one could avoid these exceptional cases and show a lower bound of $2dl$.

3.5 If d is not a Power of 2

So far we assumed that d is a power of two. Since we may always ignore some of the chips in our play, this immediately yields bounds for the general case as well. As we ignore less than half the chips, our loss is not very big. For the value of the game, we just lose an additive term of $1 + o(1)$. Unfortunately, our upper and lower bounds are already that close that such a loss is significant.

A first idea would be to partition the chips into subsets of cardinalities of powers of two, and then play the above strategies on each separately. It is a problem though to synchronize the strategies. It might happen (and Pusher cannot prevent this) that one subset already reached a Q–position ending the strategy, while another set is in the middle of a series of easy splits. To make this approach work, we would need a way to conserve the potential of a favorable position like a Q–position for several moves. This seems to be a difficult task.

Fortunately, an easy trick solves the problem and shows that the general case is not far away from the special case of powers of 2.

Lemma 8. *Let $i \in \mathbb{N}$ such that $2^i \leq d$. Then Pusher can earn a potential of $2 \cdot 2^i(i-1) \lfloor d/2^i \rfloor$. In consequence, Pusher has a strategy ensuring him a potential of at least $2d \log d(1 + o(1))$.*

Proof. Let $d_0 = \lfloor d/2^i \rfloor 2^i$, the largest multiple of 2^i not exceeding d. This is Pusher's strategy: He plays with d_0 chips only, ignoring the rest. The set of d_0 non-ignored chips is partitioned into $\lfloor d/2^i \rfloor$ groups of 2^i chips each. These groups will never be split in the course of the game, so we may assume these chips to be glued together forming 'big chips'. There are 2^i big chips, hence Pusher can follow his strategy for powers of 2 and ending up with a position of potential $2 \cdot 2^i(i-1)$ in terms of big chips. Since each big chip consists of $\lfloor d/2^i \rfloor$ ordinary ones ('solving the glue again'), this position has a potential of $2 \cdot 2^i i \lfloor d/2^i \rfloor$.

Put $l := \log d$ and $i = \lfloor l - \log l \rfloor$. Then

$$
\begin{aligned}
2 \cdot 2^i(i-1) \lfloor d/2^i \rfloor &\geq 2(d - 2^i)(l - \log l - 2) \\
&= 2dl(1 - 1/l)(1 - (\log l + 2)/l) \\
&= 2dl(1 + o(1)).
\end{aligned}
$$

□

From Lemma 1 and 8 we deduce that we know the precise value of the game, namely $\lfloor \log d + \log \log d + 1 \rfloor$, for all sufficiently large d such that the fractional part of $\log d + \log \log d$ is contained in $[0, 0.27[$. Some elementary calculus leads to the conclusion that the set of all d such that we know the precise value of the game has lower density greater than $\frac{1}{5}$.

4 Remarks and Open Problems

An obvious problem left open in this paper is a precise determination of the value of the game for all or all but a few values of d. We only succeeded in doing so for a set of d having lower density $\frac{1}{5}$. For the remaining values apart from finitely many, two possibilities exist for the value of the game.

With quite some effort it is possible to continue Pusher's strategy from the Q–positions and thus gain a potential of γdl for some $\gamma > 2$. Unfortunately, these gains are not too big, in particular, they are not enough to determine the value of the game for asymptotically all numbers d.

More interesting than a slight increase of the set of numbers d such that the value of the game with d chips is determined might be the following: Assume that $d = 2^l$ is a power of two again. Then the proofs in Section 3 give a strategy for Pusher to obtain a potential of about $2dl$. A closer inspection of these proofs shows that Pusher might need more than l^2 moves to reach this aim. This is caused by the strategy which is quite unbalanced in the following sense: Whenever Chooser has two different alternatives, i.e., Pusher did not play an easy split, one of the alternatives immediately produces a potential of $2dl$, whereas the other only gains a modest additional potential of $\Theta(d)$ in up to $l - 1$ easy splits.

We do not know whether a more balanced strategy exists. If Pusher could produce two alternatives gaining an $\Omega(d)$ potential increase in one move (like the easy splits do), this would result in a strategy that needs $O(l)$ moves only.

Acknowledgments

I would like to thank Joel Spencer for several discussions on this topic. I am also grateful to the anonymous referees for their thorough reading.

References

1. N. Alon, V. Guruswam, T. Kaufman, and M. Sudan. Guessing secrets efficiently via list decoding. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 254–262, 2002.
2. I. Barany. On a class of balancing games. *J. Combin. Theory Ser. A*, 26:115–126, 1979.
3. I. Barany and V. S. Grunberg. On some combinatorial questions in finite dimensional spaces. *Linear Algebra Appl.*, 41:1–9, 1981.
4. F. Chung, R. Graham, and T. Leighton. Guessing secrets. *Electron. J. Comb.*, 8:Research Paper 13, approx. 25 pp. (electronic), 2001.
5. F. Chung, R. Graham, and L. Lu. Guessing secrets with inner product questions. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 247–253, 2002.
6. B. Doerr. Vector balancing games respecting aging. *J. Combin. Theory Ser. A*, 95:219–233, 2001.
7. J. Olson. A balancing strategy. *J. Combin. Theory Ser. A*, 40:175–178, 1985.

8. A. Pelc. Searching games with errors — fifty years of coping with liars. *Theor. Comput. Sci.*, 270:71–109, 2002.
9. H. Peng and C. H. Yan. Balancing game with a buffer. *Adv. Appl. Math.*, 21:193–204, 1998.
10. J. Spencer. Balancing games. *J. Combin. Theory Ser. B*, 23:68–74, 1977.
11. J. Spencer. Randomization, derandomization and antirandomization: Three games. *Theor. Comput. Sci.*, 131:415–429, 1994.