# Vector Balancing Games with Aging

Benjamin Doerr*
Mathematisches Seminar II
Christian–Albrechts–Universität zu Kiel
Ludewig–Meyn–Str. 4
D–24098 Kiel, Germany
bed@numerik.uni-kiel.de

October 17, 2000

*supported by the graduate school 'Effiziente Algorithmen und Multiskalenmethoden', Deutsche Forschungsgemeinschaft

**Abstract**

In this article we study an extension of the vector balancing game investigated by Spencer and Olson (which corresponds to the on-line version of the discrepancy problem for matrices). We assume that decisions in earlier rounds become less and less important as the game continues. For an aging parameter $q \geq 1$ we define the current move to be $q$ times more important than the previous one.

We consider two variants of this problem: First, the objective is a balanced partition at the end of the game, and second, it is to ensure a balanced partition throughout the game. We concentrate on the case $q \geq 2$. We give an optimal solution for the first problem and a nearly optimal one for the second.

**Key words and phrases:** Vector balancing games, on-line algorithms, discrepancy.

# 1 Introduction and Results

## 1.1 Vector Balancing Games as On-line Version of the Discrepancy Problem

A *vector balancing problem* (also referred to as *discrepancy of matrices*) consists of a finite set $X$ of vectors and the task is to partition this set into two classes $X_1$ and $X_2$ such that the two sums $\sum_{x \in X_i} x$ , $i = 1, 2$, over all vectors in each class are ideally equal. A partition like this is called balanced. In general, of course, a perfectly balanced partition does not exist. The objective then is to minimize the imbalance $\| \sum_{x \in X_1} x - \sum_{x \in X_2} x \|$ for some norm $\| \cdot \|$.

It is convenient to represent the 2–partition by a mapping $\varepsilon : X \to \{-1, +1\}$ such that $\varepsilon(x) = 1$ holds if and only if $x \in X_1$. With this setting we can express the imbalance vector $\sum_{x \in X_1} x - \sum_{x \in X_2} x$ corresponding to the partition $(X_1, X_2)$ simply by $\sum_{x \in X} \varepsilon(x) x$.

A widely investigated special case is the discrepancy problem for hypergraphs. Here the objective is to partition the set of vertices of a hypergraph in two classes such that all hyperedges are roughly split into equal parts by this partition. By taking the column vectors of the incidence matrix as set $X$, the discrepancy problem for hypergraphs transforms into a vector balancing problem. For a deeper insight into discrepancy theory we recommend the survey of Beck and Sós [3] as well as the fourth chapter of Matoušek's recent book [7].

In this paper we focus on the on-line version of the vector balancing problem. The additional difficulty there is that one does not know the set of vectors at the beginning, but gets to know them one by one and has to decide on a sign without knowing the next vectors to come. For the analysis of this problem it is convenient to translate it into the language of games (this is a natural approach for many on-line problems). The idea is to represent both the unpredictability of the vectors and the one struggling for a balanced partition by two competing players. This yields the following two-player perfect information game:

Each round the first player (baptized 'Paul' or 'pusher' by Spencer) selects a vector $x$ from some given set $X \subseteq \mathbb{R}^d$. The second player ('Carol', 'chooser') then chooses a sign $\varepsilon \in \{-1, +1\}$ and the position vector $p$, initially set to zero, is changed to $p + \varepsilon x$. The first player's aim is to maximize $\|p\|$ for some given norm $\|\cdot\|$, while the second player tries to minimize this quantity. We call $\|p\|$ the pay-off for the first player.

A game of this kind is called a *vector balancing game* or *'pusher-chooser game'*. As mentioned, strategies for the second player correspond to the on-line algorithms for the vector balancing problem. On the other hand, strategies for the first player give lower bounds on how good an on-line algorithm can possibly be.

## 1.2 Previous Results

Several forms of vector balancing games have been studied. They differ in the set of vectors available to the first player and the norm that is used to determine the pay-off. A variant is to allow a buffer of some size where the second player can store some vectors and thus postpone the decision on the respective signs. These are some results of the different types:

*Unit Ball Games:* For a fixed norm $\|\cdot\|$ the first player may choose any vector with norm at most one, i. e. $X = \{x \in \mathbb{R}^d \mid \|x\| \leq 1\}$, and the pay-off is measured using the same norm. For the Euclidean or 2–norm it is not difficult to show that both players have strategies ensuring that $\|p\|_2 \geq \sqrt{n}$ respectively $\|p\|_2 \leq \sqrt{n}$ holds after $n$ rounds. We say that the value of this game is $\sqrt{n}$.

For the maximum norm $\|\cdot\|_\infty$, Spencer [10] gave an upper bound of $\sqrt{2n\ln(2d)}$. For the $d$ round game he proved a lower bound of $\sqrt{d \log d}(1 - o(1))$ in [11].

*Discrete Games:* For games with finite set $X$, Barany [1] found a complete solution. His result implies that in the case $X = \{0, 1\}^d$ and $\|\cdot\| = \|\cdot\|_\infty$ the value of the game played sufficiently many rounds is $2^{d-2}$.

*Games with Buffer:* The first result allowing a buffer is due to Barany and Grunberg [2]. They show that given a buffer of size $d$ the second player can keep $\|p\|$ below $2d$ no matter what norm is used (the same norm is required in the definition of $X = \{x \in \mathbb{R}^d \mid \|x\| \leq 1\}$ and the pay-off). This was improved by Peng and Yan [9], who proved that a buffer of size $d-1$ already suffices. They also remark that for the 2–norm allowing a buffer of less than $d-1$ vectors gives no improvement compared to the no-buffer case.

### 1.3 Our Contribution

All games cited above relate to on-line problems without temporal aspects. By this we mean that a decision is the same important throughout the game. In this article we will assume that a decision made in the past (i. e. in an earlier round of the game) is less important than a newer one. This is represented by the different update rule $p := \frac{1}{q}p + \varepsilon x$ for some aging parameter $q > 1$. Hence the current decision is $q$ times more important than the preceding one. We restrict ourselves to the maximum norm unit ball game, that is, the first player selects vectors from $\{x \in \mathbb{R}^d \mid \|x\|_\infty \leq 1\}$, and the pay-off is measured using $\|\cdot\|_\infty$ on $p$ as well. Since we will not need any other norms let us agree that from now on $\|\cdot\|$ shall always denote the maximum norm $\|\cdot\|_\infty$.

Immediately we see that the pay-off is bounded by $\frac{q}{q-1}$. This is due to the update rule which rescales the importance of decisions in the past relative to the actual one. A different approach working with absolute values is the following: In round $i$ the first player chooses a vector $x^{(i)}$ with norm at most $q^{i-1}$ and the the second player updates the position vector either to $p := p + x^{(i)}$ or $p := p - x^{(i)}$. The values of an $n$ round game then differs from our approach by a factor of $q^{n-1}$. Hence we lose nothing by investigating the first approach which we find more natural.

The nature of the game is different depending on whether the aging parameter is at least 2 or not. In the first case the aging aspect is dominant. The strategies are completely different from the ones in the game without aging, that is $q = 1$. For $1 < q < 2$ the aging is less important. This case requires strategies different from the non-aging case and the pure aging case with $q \geq 2$. Therefore, we restrict ourselves to the case $q \geq 2$ in this paper.

Contrary to the no-buffer games described in the previous section, in our setting there are reasonable strategies such that the maximum value for $\|p\|$ does not necessarily occur after the last round. This motivates the distinction of two versions of the game: First, the value of $\|p\|$ after the last round is the pay-off for the first player, and second, the maximum value of $\|p\|$ occurred during the game is the pay-off for the first player. We call the two versions the *fixed end version* and *continuous version* respectively. The second version also refers to the case that the game is played for a fixed number of rounds which is not known to the second player[1].

We show that the fixed end version of the game has value $\frac{q-q^{-\lfloor \log_2 d \rfloor}}{q-1}$ if at least $\log_2 d + 1$ rounds are played (otherwise it is $\frac{q-q^{-r+1}}{q-1}$, where $r$ denotes the number of rounds). Note that the number $\frac{q-q^{-\lfloor \log_2 d \rfloor}}{q-1}$ is the maximum imbalance that can occur in a $\lfloor \log_2 d \rfloor + 1$ round game by putting all vectors into the same partition class. We may thus interpret our result like this: The optimal strategy for the second player in the fixed end vector balancing game leads to a partition which is perfect apart from the last $\lfloor \log_2 d \rfloor + 1$ vectors. This seems to be a more intuitive way of stating the result. Let us therefore define

$$v_q(r) := \frac{q - q^{-r+1}}{q-1},$$

the maximum imbalance that can occur in an $r$ round game by putting all vectors into the same partition class.

For the continuous version we show that the first player can get a pay-off of at least $\frac{q-2q^{-\lfloor \log_2 d \rfloor - \lfloor \log_2 \log_2 d \rfloor + 1}}{q-1}$ (again assuming sufficiently many rounds played), while the second player has a strategy keeping $\|p\|$ below $\frac{q}{q-1} - q^{-\log_2 d - \log_2 \log_2 d - 4}$ throughout the game. In particular, the value $v$ of this game satisfies

$$v_q(\lfloor \log_2 d \rfloor + \lfloor \log_2 \log_2 d \rfloor - 1) \leq v \leq v_q(\log_2 d + \log_2 \log_2 d + 5).$$

We see that the continuous problem is significantly harder than the fixed end version (this also applies to their respective analyses).

The fixed end version with $q = 2$ has a nice theoretical application. In [5] it

[1] Actually, Spencer's proof for the upper bound in [10] also requires the second player to know the number of rounds. Olson [8] later gave a strategy that does not need this information and still yields the same bound (apart from constants).

is used to prove that

$$\begin{aligned} \operatorname{lindisc}(A) &\leq 2\left(1 - 2^{-\lfloor \log_2(m) \rfloor - 1}\right) \operatorname{herdisc}(A) \\ ( &\leq 2\,(1 - \tfrac{1}{2m})\operatorname{herdisc}(A)\ ) \end{aligned}$$

holds for any matrix $A \in \mathbb{R}^{m \times n}$. This improves an earlier result in this direction by Lovász et al. [6] and Beck and Spencer [4], and is a step towards Spencer's conjecture $\operatorname{lindisc}(A) \leq 2(1 - \frac{1}{n+1})\operatorname{herdisc}(A)$.

# 2 The Fixed End Problem

In this section we analyze the version of the vector balancing game with aging where the pay-off for the first player is the value of $\|p\|$ after a fixed number $n$ of rounds. The objective of the players thus is an imbalanced (respectively a balanced) partition at the end of an $n$ round game. Let us fix the rules of this game, which we denote by $G_{ndq}$:

Initially the position vector $p \in \mathbb{R}^d$ is zero. A round of the game consists of three steps:

(i) Player $A$ selects a vector $x \in \mathbb{R}^d$ such that $\|x\| \leq 1$,

(ii) Player $B$ chooses a sign $\varepsilon \in \{-1, +1\}$,

(iii) the position vector is updated to $p := \frac{1}{q}p + \varepsilon x$.

The game is played for $n$ rounds. The value $\|p\|$ at the end of the game is the pay-off for player $A$, i. e. Player $A$ aims to maximize $\|p\|$ and Player $B$ to minimize this quantity. The maximum pay-off Player $A$ can enforce is called the value $v(G_{ndq})$ of the game.

In the following let us assume that $q \geq 2$. In the analysis of this case (see the proof of Theorem 1 below) we exhibit a surprising phenomenon. It turns out that mainly the last $\lfloor \log_2 d \rfloor + 1$ moves are important. For this reason the players need to know the number of rounds. Any value $\|p\|$ that Player $A$ might have reached up to round $n - \lfloor \log_2 d \rfloor - 1$ will not only not help him, but even be contraproductive. Hence up to this point the players will pursue the opposite aims. The optimal strategy for player $A$ is to select $x^{(i)} = 0$ for all but the last $\lfloor \log_2 d \rfloor + 1$ moves, thus minimizing $\|p\|$ in this stage of the game.

**Theorem 1.** *Assume $q \geq 2$ and let $n, d \in \mathbb{N}$. Set $r := \min\{n, \lfloor \log_2 d \rfloor + 1\}$. The value of the game $G_{ndq}$ is*

$$v(G_{ndq}) = \tfrac{q-q^{-r+1}}{q-1} = v_q(r).$$

*Proof.* Player $A$ has the following strategy: Choose the first $n - r$ vectors as zero ($x^{(i)} := 0$ for all $i \in [n - r]$). The last $r$ vectors choose like this: Components with index greater than $2^r$ are always set zero (for instance). For an index $i = 1+\sum_{j=0}^{r-1} a_j 2^j \leq 2^r$, $a_0, \ldots, a_{r-1} \in \{0, 1\}$ and a $p \in \{0, \ldots, r-1\}$ set $x_i^{(n-p)} := 2a_p - 1$. Here is an example for $d = 5$:

$$\begin{aligned} x^{(j)} &= (0, 0, 0, 0, 0) \text{ for } i \in [n-3] \\ x^{(n-2)} &= (-1, -1, -1, -1, 0) \\ x^{(n-1)} &= (-1, -1, +1, +1, 0) \\ x^{(n)} &= (-1, +1, -1, +1, 0). \end{aligned}$$

Whatever signs $\varepsilon^{(j)} \in \{-1, +1\}, j \in [n]$ are chosen, there will always be an index $i \in [2^r]$ such that $x_i^{(n-r+1)} = \cdots = x_i^{(n)} \in \{-1, +1\}$. From

$$\sum_{j=1}^{n} q^{-n+j} x_i^{(j)} = x_i^{(n)} \sum_{j=0}^{r-1} q^{-j} = x_i^{(n)} \frac{1 - q^{-r}}{1 - \frac{1}{q}}$$

we conclude that the value of the game is at least $\frac{q-q^{-r+1}}{q-1}$.

This bound is sharp, as the following strategy for Player $B$ reveals. If $r = n$, then any choice of signs $\varepsilon^{(1)}, \ldots, \varepsilon^{(r)}$ by Player $B$ keep the pay-off below our claimed value. Hence let us assume $r < n$. Whatever vectors $x^{(1)}, \ldots, x^{(n-r)}$ Player $A$ chooses in the first $n - r$ rounds, pick $\varepsilon^{(1)}, \ldots, \varepsilon^{(n-r)} := +1$ (any other choice would do, too). Set $p := \sum_{j=1}^{n-r} q^{-n+j} x^{(j)}$. Choose the next sign $\varepsilon^{(n-r+1)} \in \{-1, +1\}$ in such a way that the number of indices $i \in X_1 := \{i \in [d] | p_i \neq 0\}$ such that $\operatorname{sgn}(p_i)$ and $\operatorname{sgn}(\varepsilon^{(k-r+1)} x_i^{(k-r+1)})$ are different is maximal. Set

$$X_2 := \{i \in X_1 \mid \operatorname{sgn}(p_i) = \operatorname{sgn}(\varepsilon^{(k-r+1)} x_i^{(n-r+1)})\}.$$

Next choose $\varepsilon^{(n-r+2)} \in \{-1, +1\}$ such that the number of indices $i \in X_2$ such that $\operatorname{sgn}(p_i)$ and $\operatorname{sgn}(\varepsilon^{(n-r+2)} x_i^{(n-r+2)})$ are different is maximal. Set

$$X_3 := \{i \in X_2 \mid \operatorname{sgn}(p_i) = \operatorname{sgn}(\varepsilon^{(n-r+2)} x_i^{(n-r+2)})\}.$$

Continue in this fashion until $\varepsilon^{(n)}$ and $X_r$ are determined.

Note that

$$|X_{j+1}| \leq \left\lfloor \tfrac{1}{2} |X_j| \right\rfloor$$

for all $j \in [r-1]$. From $|X_1| \leq d$ we conclude $|X_r| < 1$, that is $X_r = \emptyset$. So for every index $i \in [d]$ there is a $j \in \{n-r+1, \ldots, n\}$ such that $\operatorname{sgn}(p_i) \neq \operatorname{sgn}(\varepsilon^{(j)} x_i^{(j)})$, or $p_i = 0$. The worst case — and here $q \geq 2$ comes into play — is the one where for one index $i \in [d]$ all $\varepsilon^{(j)} x_i^{(j)}$, $j \in \{n-r+1, \ldots, n\}$ are 1 (or $-1$) and $p_i$ is zero. For the pay-off at the end of the game we thus have

$$\begin{aligned}
\left\| \sum_{j=1}^{n} q^{-n+j} \varepsilon^{(j)} x^{(j)} \right\| &= \left\| \sum_{j=n-r+1}^{n} q^{-n+j} \varepsilon^{(j)} x^{(j)} + p \right\| \\
&\leq \sum_{z=0}^{r-1} q^{-z} \\
&= \frac{1 - q^{-r}}{1 - \frac{1}{q}}.
\end{aligned}$$

This ends the proof. □

## 3 The Continuous Problem

In this section we investigate the version of the problem where we want to ensure a balanced partition throughout the game. We consider the case that the game is played on and on, and that the pay-off for Player $A$ is the supremum over all values of $\|p\|$ that occurred during play. From the viewpoint of the second player, this is equivalent to saying that the game $G_{ndq}$ is played with the additional restriction that he does not know the number of rounds. We denote this game by $G_{\infty dq}$.

A particular sequence of moves respecting the rules of the game shall be called an instance of the game. Formally, it is a pair $I = ((x^{(i)})_{i \in \mathbb{N}}, (\varepsilon^{(i)})_{i \in \mathbb{N}})$

such that $\|x^{(i)}\| \leq 1$ and $\varepsilon^{(i)} \in \{-1, +1\}$ for all $i \in \mathbb{N}$. From the definition we see that the pay-off for this instance is

$$\sup_{n \in \mathbb{N}} \sum_{i \in [n]} q^{n-i} \varepsilon^{(i)} x^{(i)}.$$

Let us assume again $q \geq 2$. This will not be necessary for the main ideas of the proof, but the result gets less interesting for $q$ close to 1. We show

**Theorem 2.** *The value of the game $G_{\infty dq}$ is bounded by*

$$\tfrac{q - 2q^{-\lfloor \log_2 d \rfloor - \lfloor \log_2 \log_2 d \rfloor + 1}}{q-1} \leq v(G_{\infty dq}) \leq \tfrac{q}{q-1} - q^{-\log_2 d - \log_2 \log_2 d - 4}.$$

*In particular,*

$$v_q(\lfloor \log_2 d \rfloor + \lfloor \log_2 \log_2 d \rfloor - 1) \leq v(G_{\infty dq}) \leq v_q(\log_2 d + \log_2 \log_2 d + 5).$$

Similarly to the fixed end version of the game we will also work with strategies of 'changing signs'. A first strategy for player $B$ is to enforce a 'change of signs' in every block $B_k$ of rounds $(k-1)(\lfloor \log_2 d \rfloor + 1) + 1, \ldots, k(\lfloor \log_2 d \rfloor + 1)$. This is equivalent to saying that Player $B$ should play according to his strategy of section 2 assuming all rounds $k(\lfloor \log_2 d \rfloor + 1), k \in \mathbb{N}$ to be last rounds. In the worst case, this might lead to a subsequence of rounds $S = \{(k-1)(\lfloor \log_2 d \rfloor + 1) + 2, \ldots, (k+1)(\lfloor \log_2 d \rfloor + 1) - 1\}$ such that for some index $j \in [d]$ all the $2 \lfloor \log_2 d \rfloor$ values $\varepsilon^{(i)} x_j^{(i)}, i \in S$ have the same sign different from zero.

We show that Player $B$ can do much better: He has a strategy such that for every index $j \in [d]$ and any set $S$ of at least $\log_2 d + \log_2 \log_2 d + 5$ successive rounds there are $i_1, i_2 \in S$ such that $\operatorname{sgn}(\varepsilon^{(i_1)} x_j^{(i_1)}) \neq \operatorname{sgn}(\varepsilon^{(i_2)} x_j^{(i_2)})$ or $x_j^{(i_1)} = x_j^{(i_2)}$. This bounds the value of the game $G_{\infty dq}$ as the next lemma shows.

**Lemma 1.** *Let $r \in \mathbb{N}$.*

*(i) Suppose that for every sequence $(x^{(k)})_{k \in \mathbb{N}}$ of vectors Player $B$ can choose his moves $(\varepsilon^{(k)})_{k \in \mathbb{N}}$ in such a way that for every $k \in \mathbb{N}$ and every index $i \in [d]$ not all signs $\operatorname{sgn}(\varepsilon^{(k)} x_i^{(k)}), \ldots, \operatorname{sgn}(\varepsilon^{(k+r)} x_i^{(k+r)})$ are equal and different from zero.*

*Then $v(G_{\infty dq}) \leq \frac{q}{q-1} - q^{-r} \leq v_q(r+1)$.*

(ii) *Suppose that Player A has a strategy using* $-1, 1$ *vectors such that there exist* $k \in \mathbb{N}$ *and* $i \in [d]$ *such that all numbers* $\varepsilon^{(k)} x_i^{(k)}, \ldots, \varepsilon^{(k+r-1)} x_i^{(k+r-1)}$ *are equal.*

*Then* $v(G_{\infty dq}) \geq \frac{q - 2q^{-r+1}}{q-1} \geq v_q(r-1)$.

*Proof.* Let $(x^{(k)})_{k \in \mathbb{N}}$ be a sequence of vectors in $[-1, 1]^d$ and $(\varepsilon^{(k)})_{k \in \mathbb{N}}$ be a sequence in $\{-1, +1\}$.

Assume first that for every $k \in \mathbb{N}$ and every index $i \in [d]$ not all numbers $\varepsilon^{(k)} x_i^{(k)}, \ldots, \varepsilon^{(k+r)} x_i^{(k+r)}$ have the same sign different from zero. Let $k > r$ and $p := \sum_{j \in [k]} q^{-k+j} \varepsilon^{(j)} x^{(j)}$ denote the position vector at the end of round $k$. Let $i \in [d]$. As not all of the numbers $\varepsilon^{(k-r)} x_i^{(k-r)}, \ldots, \varepsilon^{(k)} x_i^{(k)}$ have the same sign, $|p_i|$ is maximal if $x_i^{(k-r)}$ is zero and other numbers $\varepsilon^{(j)} x_i^{(j)}, j \in \{k-r+1, \ldots, k\}$ are all 1 or all $-1$. Thus we have

$$|p_i| = \left| \sum_{j \in [k]} q^{-k+j} \varepsilon^{(j)} x_i^{(j)} \right| \leq \sum_{\substack{j \in [k] \\ j \neq k-r}} q^{-k+j} < \frac{q}{q-1} - q^{-r}.$$

Assume now that all vectors are $-1, 1$ vectors and there are $k \in \mathbb{N}$ and an index $i \in [d]$ such that all numbers $\varepsilon^{(k)} x_i^{(k)}, \ldots, \varepsilon^{(k+r-1)} x_i^{(k+r-1)}$ are equal. Let $p := \sum_{j \in [k+r-1]} q^{-k-r+1+j} \varepsilon^{(j)} x^{(j)}$ denote the position vector at the end of round $k + r - 1$. We have

$$\begin{aligned}
|p_i| &= \left| \sum_{j \in [k+r-1]} q^{-k-r+1+j} \varepsilon^{(j)} x_i^{(j)} \right| \\
&\geq \left| \sum_{j=k}^{k+r-1} q^{-k-r+1+j} \varepsilon^{(j)} x_i^{(j)} \right| - \left| \sum_{j=1}^{k-1} q^{-k-r+1+j} \varepsilon^{(j)} x_i^{(j)} \right| \\
&\geq \sum_{j=0}^{r-1} q^{-j} - q^{-r} \sum_{j=0}^{k-2} q^{-j} \\
&> \frac{1 - q^{-r}}{1 - \frac{1}{q}} - q^{-r} \frac{1}{1 - \frac{1}{q}} = \frac{q - 2q^{-r+1}}{q-1}.
\end{aligned}$$

□

Note that an upper bound of $\frac{q}{q-1} - \frac{2q^{-r+1}}{q-q^{-r}}$ can be shown by applying the assumption of changing signs not only on the last $r+1$ vectors, but on every group of $r+1$ vectors. A similar argument improves the lower bound slightly.

By Lemma 1 the analysis of $G_{\infty dq}$ is reduced to sign changing strategies. Ignoring for the moment the possibility that some $x_i^{(r)}$ might be zero, we notice: By choosing $\varepsilon^{(r)} = 1$, a change of sign is inflicted on all those components $i$ such that $\operatorname{sgn}(x_i^{(r)}) \neq \operatorname{sgn}(\varepsilon^{(r-1)} x_i^{(r-1)})$, while choosing $\varepsilon^{(r)} = -1$ yields a change of sign on the opposite components. We may therefore investigate a simplified game $C_d$ (which in particular is independent of $q$): Given are $d$ piles $p_1, \dots, p_d$ of, say, cards that initially hold one card each. A round of this game consists of the three steps

(i) Player $A$ selects a set $S \subseteq [d]$ of piles,

(ii) Player $B$ either removes all cards from the piles in $S$ or all cards from the piles in $[d] \setminus S$. Formally, $B$ chooses a set $T \in \{S, [d] \setminus S\}$ and resets $p_i := 0$ for all $i \in T$.

(iii) One card is placed on every pile ($p_i := p_i + 1$ for all $i \in [d]$).

The game is played infinitely many rounds. The pay-off for Player $A$ is the maximum value of $\|p\|$ that occurred during play. This game is similar to the tenure game Spencer investigated in [12]. Instead of step (iii), there a card is added only to those piles $p_i$ that have $p_i \neq 0$. Hence the number of active piles reduces in play and finally is zero. The tenure game has a nice solution: Both players do their best if they choose their moves in such a way that the potential function $\sum_{i \in [d]} 2^i$ is maximized respectively minimized. For the game $C_d$ a solution like this can not be expected as all piles keep active throughout the game.

Before analyzing the game $C_d$ let us first fix the connection with $G_{\infty dq}$:

**Lemma 2.** *Suppose that the value of $C_d$ is $r$, i. e. Player $B$ has a strategy such that no pile ever contains more than $r$ cards and Player $A$ can enforce a pile of height $r$. Then Player $B$ has a strategy as in Lemma 1(i) and Player $A$ has a strategy as in Lemma 1(ii). In particular, the value of $C_d$ determines the value of $G_{\infty dq}$ almost completely:*

$$v_q(v(G_d) - 1) \leq v(G_{\infty dq}) \leq v(v(G_d) + 1).$$

*Proof.* Let $I_G = ((x^{(j)})_{j\in\mathbb{N}}, (\varepsilon^{(j)})_{j\in\mathbb{N}})$ be an instance of the game $G_{\infty dq}$. We call an instance $I_C = ((S^{(j)})_{j\in\mathbb{N}}, (T^{(j)})_{j\in\mathbb{N}})$ of $C_d$ corresponding to $I_G$ if

(i) $\forall j \in \mathbb{N} : S^{(j)} = \{i \in [d] \,|\, \operatorname{sgn}(\varepsilon^{(j)} x_i^{(j)}) = \operatorname{sgn}(x_i^{(j+1)}) \neq 0\}$,

(ii) $\forall j \in \mathbb{N} : T^{(j)} = S^{(j)} \iff \varepsilon^{(j+1)} = -1$.

In particular $I_C$ is uniquely determined by $I_G$.

Suppose $I_G$ and $I_C$ as above and corresponding. Then for all $i \in [d]$ and $k \in \mathbb{N}$ we have

$$i \in T^{(k)} \iff \left|\operatorname{sgn}(\varepsilon^{(k)} x_i^{(k)}) + \operatorname{sgn}(\varepsilon^{(k+1)} x_i^{(k+1)})\right| \leq 1.$$

Note that the right-hand side just means that $\varepsilon^{(k)} x_i^{(k)}$ and $\varepsilon^{(k+1)} x_i^{(k+1)}$ have different signs or are both zero. Hence for an index $i \in [d]$ and $r \in \mathbb{N}$ we have that all numbers $\varepsilon^{(k)} x_i^{(k)}, \ldots, \varepsilon^{(k+r-1)} x_i^{(k+r-1)}$ have the same non-zero sign if and only if $i \notin T^{(k)} \cup \ldots \cup T^{(k+r-2)}$. This is equivalent to the fact that the position vector $p^{(k+r-2)}$ after round $k+r-2$ in $C_d$ fulfills $p_i^{(k+r-2)} \geq r$. Thus it is enough to show that Player $B$ can choose signs in game $G_{\infty dq}$ such that the corresponding instance of $C_d$ has value at most $r$ and Player $A$ can choose $\{-1, 1\}$ vectors such that the corresponding instance of $C_d$ has value at least $r$.

Player $B$'s strategy for $C_d$ yields the following strategy for the game $G_{\infty dq}$. For the first move in $G_{\infty dq}$ Player $B$ may choose any sign $\varepsilon^{(1)}$. After $A$'s second move $x^{(2)}$ set $S^{(1)} = \{i \in [d] \,|\, \operatorname{sgn}(\varepsilon^{(1)} x_i^{(1)}) = \operatorname{sgn}(x_i^{(2)}) \neq 0\}$. Choose $T^{(1)} \in \{S^{(1)}, [d] \backslash S^{(1)}\}$ according to the strategy that keeps the position vector in $C_d$ at norm at most $r$. If $T^{(1)} = S^{(1)}$ select $\varepsilon^{(2)} = -1$ in $G_{\infty dq}$, and $\varepsilon^{(2)} = 1$ otherwise. Continue like this for all rounds of the game. It is clear that $I_G = ((x^{(j)})_{j\in\mathbb{N}}, (\varepsilon^{(j)})_{j\in\mathbb{N}})$ and $I_C = ((S^{(j)})_{j\in\mathbb{N}}, (T^{(j)})_{j\in\mathbb{N}})$ are corresponding instances such that at no time any pile in $I_C$ gets higher than $r$.

The following strategy serves for Player $A$. For the first move in $G_{\infty dq}$ Player $A$ may choose any $\{-1, +1\}$ vector $x^{(1)}$. Set $p = (1, \ldots, 1)^\top \in \mathbb{R}^d$. Let $S^{(1)}$ be a choice of $A$ in $C_d$ following the strategy that enforces a pile of height $r$ once in the game $C_d$. Define $x^{(2)} \in \mathbb{R}^d$ by

$$x_i^{(2)} := \begin{cases} \varepsilon^{(1)} x_i^{(1)} & \text{if } i \in S \\ -\varepsilon^{(1)} x_i^{(1)} & \text{else} \end{cases} .$$

If Player $B$ chooses $\varepsilon^{(2)} = -1$, set $T^{(1)} = S^{(1)}$, else set $T^{(1)} = [d]\backslash S^{(1)}$. Update the position vector $p$ of the card game as required by the rules. Continue like this for the rest of the game. As Player $A$ is following his strategy for $C_d$, there will once be a pile of height $r$. By definition these instances of the vector game and the card game are corresponding. Thus there are $k \in \mathbb{N}$, $i \in [d]$ such that all numbers $\varepsilon^{(k)}x_i^{(k)}, \ldots, \varepsilon^{(k+r-1)}x_i^{(k+r-1)}$ have the same sign different from zero. This shows that $A$'s strategy for $G_{\infty dq}$ is as required by Lemma 1(ii). $\square$

To complete the proof of Theorem 2 we bound the value $v(C_d)$ of the card game $C_d$.

**Lemma 3.** *For $d \geq 3$ the value of the card game $C_d$ satisfies*

$$\begin{aligned} \lfloor \log_2 d \rfloor + \lfloor \log_2 \log_2 d \rfloor \leq v(C_d) &\leq \frac{\log_2 d + \log_2 \log_2 d}{\log_2 \left(2 - \frac{1}{\log_2 d}\right)} + 1 \\ &\leq \log_2 d + \log_2 \log_2 d + 4. \end{aligned}$$

*For $d = 2$ we have $v(C_d) = 2$.*

*Proof.* The case $d = 2$ is solved by a moment's thought, so let us assume $d \geq 3$. We first observe that the order of the piles is irrelevant, hence we may describe the actual position vector by an expression $x_1^{n_1} x_2^{n_2} \ldots x_l^{n_l}$ meaning that there are $n_i$ piles each holding $x_i$ cards for all $i \in [l]$. Similarly, a move by Player $A$ (which is a subset $S$ of the index set $[d]$) can be described by such an expression (again it is not important which of the possibly several piles of same size are in $S$).

We start with a strategy for Player $A$. Let us assume first that $d$ is a power of 2. It is clear that Player $A$ can change a position vector $x_1^{n_1} x_2^{n_2} \ldots x_l^{n_l}$, where all $n_i, i \in [l]$ are even, to the position vector $(x_1+1)^{\frac{n_1}{2}} (x_2+1)^{\frac{n_2}{2}} \ldots (x_l+1)^{\frac{n_l}{2}} 1^{\frac{d}{2}}$ by selecting the set $x_1^{\frac{n_1}{2}} x_2^{\frac{n_2}{2}} \ldots x_l^{\frac{n_l}{2}}$. Repeated application of this strategy on the initial position vector of $1^d$ leads to the position $(\log_2 d + 1)^1 (\log_2 d)^1 (\log_2 d - 1)^2 (\log_2 d - 2)^4 \ldots 2^{\frac{d}{4}} 1^{\frac{d}{2}}$.

We call a partial position (which is just the restriction of $p$ to a subset of $[d]$) a logarithmic ladder of type $L(s, e, l)$ for some $s, e, l \in \mathbb{N}, s \geq l$, if it equals $s^e(s-1)^{2e}(s-2)^{4e}\dots(s-l+1)^{2^{(l-1)}e}$. In this notation we just showed that Player $A$ can enforce a logarithmic ladder $L(\log_2 d, 1, \log_2 d)$. We now show that Player $A$ can enforce a logarithmic ladder $L(s+1, 1, \lfloor \frac{l}{2} \rfloor)$ from a position containing a logarithmic ladder $L(s, 1, l)$. $A$'s first move is $S = L(s, 1, \lfloor \frac{l}{2} \rfloor)$. If $B$ chooses $S$, then (among other piles) a logarithmic ladder $L(s - \lfloor \frac{l}{2} \rfloor, 2^{\lfloor \frac{l}{2} \rfloor}, \lceil \frac{l}{2} \rceil)$ remains and is updated to $L(s+1-\lfloor \frac{l}{2} \rfloor, 2^{\lfloor \frac{l}{2} \rfloor}, \lceil \frac{l}{2} \rceil)$ in step (iii) of this round. By the equi-partition argument from above (applied $\lfloor \frac{l}{2} \rfloor$ times) Player $A$ can enforce a logarithmic ladder $L(s+1, 1, \lceil \frac{l}{2} \rceil)$. On the other hand, if $B$ chooses the complement of $S$, then $S$ is simply updated to $L(s+1, 1, \lfloor \frac{l}{2} \rfloor)$.

Applying this logarithmic ladder partition strategy $\lfloor \log_2 \log_2 d \rfloor$ times on $L(\log_2 d, 1, \log_2 d)$, Player $A$ can reach a logarithmic ladder $L(\log_2 d + \lfloor \log_2 \log_2 d \rfloor, 1, 1)$, which is nothing more than a single pile of height $\log_2 d + \lfloor \log_2 \log_2 d \rfloor$. This proves the lower bound for $d$ a power of 2. If $d$ is not a power of 2, Player $A$ fixes a set of $2^{\lfloor \log_2 d \rfloor}$ piles, plays on these piles according to the strategy just described and ignores the remaining piles.[2]

For the upper bound set $\lambda := \frac{2\log_2 d - 1}{\log_2 d}$ and $v : \mathbb{N} \to \mathbb{R}; i \mapsto \lambda^{i-1}$. Set $v(p) := \sum_{i \in [d]} v(p_i)$ for a position vector $p \in \mathbb{N}^d$. We analyze the strategy for Player $B$ to choose that one of the alternatives which minimizes $v(p)$ for the resulting position vector $p$. Write $p \circ T$ for the position vector resulting from $p$ if Player $B$ chooses the set $T$ of piles to be emptied. We have

$$v(p \circ T) = |T|v(1) + \sum_{i \in [d] \setminus T} v(p_i + 1) = |T| + \lambda \sum_{i \in [d] \setminus T} v(p_i).$$

We claim that Player $B$ can ensure $v(p) \leq d \log_2 d$ throughout the game. This is clear for the start (where $p_i = 1$ for all $i \in [d]$), so let us assume that we are in some round such that the position vector $p$ at the start of this

[2] Joel Spencer (private communication, 2000) noted that the last two paragraphs can be replaced by a potential function argument. Having reached the position $(\log_2 d + 1)^1 (\log_2 d)^1 (\log_2 d - 1)^2 (\log_2 d - 2)^4 \dots 2^{\frac{d}{4}} 1^{\frac{d}{2}}$, Player $A$ may use his strategy from the tenure game. This means trying to maximize the potential function $v(p) = \sum_{i \in [d]} 2^{p_i}$. Doing so involves the so-called splitting lemma which states that he can split the piles into two groups having similar potential. See [12] for the details. As a result of this approach, the lower bound improves to $\lfloor \log_2 d + \log_2(2 + \log_2 d) \rfloor$. This is not so important for our purposes, but both beautiful and a great step towards the determination of the exact value of this game.

round fulfills $v(p) \le d \log_2 d$. Let $S \subseteq [d]$ denote $A$'s move and $T$ be one of the alternatives $S$ and $[d] \setminus S$ which minimizes $v(p \circ T)$. Then

$$\begin{aligned} v(p \circ T) &\le \tfrac{1}{2}(v(p \circ S) + v(p \circ ([d] \setminus S))) \\ &= \tfrac{1}{2}(|S| + \sum_{i \in [d] \setminus S} \lambda v(p_i) + |[d] \setminus S| + \sum_{i \in S} \lambda v(p_i)) \\ &= \tfrac{1}{2}(d + \lambda v(p)) \\ &\le \tfrac{1}{2}(d + \frac{2 \log_2 d - 1}{\log_2 d} d \log_2 d) \\ &= d \log_2 d. \end{aligned}$$

Hence $B$'s strategy ensures that $v(p) \le d \log_2 d$ holds throughout the game. This implies

$$\lambda^{p_i - 1} = v(p_i) \le v(p) \le d \log_2 d$$

for all $i \in [d]$. Hence

$$\|p\| \le \log_\lambda (d \log_2 d) + 1 = \frac{\log_2 d + \log_2 \log_2 d}{\log_2 \left(2 - \frac{1}{\log_2 d}\right)} + 1 \le \log_2 d + \log_2 \log_2 d + 4,$$

where the last inequality follows from some calculus. □

The proof of Theorem 2 follows from Lemma 1 to 3.

## 4 Summary and Outlook

This paper is a first exposition of vector balancing problems that contain a temporal aspect. For a specific aging assumption we gave optimal respectively quasi-optimal strategies for the two different problems that a balanced partition is required at the end of the game and throughout the game.

There are many balancing problems with some aging aspect thinkable. We believe that the one we chose is one of the more interesting ones. Let us briefly analyze what happens if we add the aging aspect to the problems mentioned in the introduction:

For the $\|\cdot\|_2$–norm problem the non-aging case simply generalizes to

**Theorem 3.** *Let $q \geq 1$. The vector balancing game $G_{ndq}$ where the maximum norm is replaced with the Euclidean norm has value*

$$\sqrt{\sum_{i=0}^{n-1} \frac{1}{q^{2i}}}.$$

*It makes no difference whether the players know the number of rounds or not.*

Allowing a buffer seems a little strange. On the one hand time plays an important role (aging) and on the other decisions can be postponed (buffer concept). Some results also give the impression that buffers are not very helpful: For the fixed end version $G_{ndq}$ with $q \geq 2$, it is easy to see that a buffer does not change the value of the game. This also leads to a lower bound for the continuous version. Similarly, for the Euclidean game of Theorem 3 a buffer of size less than $d-1$ gives no improvement. For a buffer of any size $\sqrt{\sum_{i=0}^{d-1} \frac{1}{q^{2i}}}$ is a lower bound. Hence for $q$ not too close to 1, the effect of a buffer is small.

Discrete vector balancing games (i. e. the first player may choose his vectors from a finite set) are harder to analyze. As our fooling strategies used $-1, 0, +1$ vectors only, our lower bounds are still valid for $X = \{-1, 0, +1\}^d$ (and the upper ones all the more). For finite sets $X$ different from that, the problems seems to depend heavily on the individual structure of $X$.

A completely different problem arises from a linear aging assumption, that is, the position vector at the end of round $k$ is defined by

$$\sum_{i=0}^{\min\{l,k\}-1} (1 - \tfrac{i}{l}) \varepsilon^{(k-i)} x^{(k-i)}$$

for some parameter $l \in \mathbb{N}$. For small $l$, that is fast aging, our sign changing argument is sufficient again. For the fixed end version ($n$ rounds) and $l \leq \lfloor \log_2 d \rfloor + 1$, the first player can enforce a position vector $p$ satisfying $\|p\|_\infty = \sum_{i=0}^{\min\{l,n\}-1} (1 - \frac{i}{l})$ by the same strategy he used in Section 2. Similarly, in the open end version he can enforce $\|p\|_\infty = \sum_{i=0}^{l-1} (1 - \frac{i}{l})$ if $l \leq v(C_d)$. Clearly, both of these results are optimal. For larger values of $l$ the problem is open.

A second open problem is the case $1 < q < 2$ of the games $G_{ndq}$ and $G_{\infty dq}$.

## Acknowledgments

I would like to thank Joel Spencer for drawing my interest on vector balancing games and several discussions on this topic.


## References


[1] I. Barany. On a class of balancing games. *J. Comb. Theory, Ser. A*, 26:115–126, 1979.

[2] I. Barany and V. S. Grunberg. On some combinatorial questions in finite dimensional spaces. *Linear Algebra Appl.*, 41:1–9, 1981.

[3] J. Beck and V. Sós. Discrepancy theory. In R. Graham, M. Grötschel, and L. Lovász, editors, *Handbook of Combinatorics*. 1995.

[4] J. Beck and J. Spencer. Integral approximation sequences. *Math. Programming*, 30:88–98, 1984.

[5] B. Doerr. Linear and hereditary discrepancy. Technical Report 98–34, Berichtsreihe des Mathematischen Seminars der Universität Kiel, 1998. 7 pages. Available at `http://www.numerik.uni-kiel.de/reports/1998/98-34.ps.gz`. Accepted for publication in *Combinatorics, Probability and Computing*.

[6] L. Lovász, J. Spencer, and K. Vesztergombi. Discrepancies of set-systems and matrices. *European J. Combin.*, 7:151–160, 1986.

[7] J. Matoušek. *Geometric Discrepancy*. Springer, 1999.

[8] J. Olson. A balancing strategy. *J. Comb. Theory, Ser. A*, 40:175–178, 1985.

[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. Comb. Theory, Ser. B*, 23:68–74, 1977.

[11] J. Spencer. *Ten Lectures on the Probabilistic Method*. SIAM, 1987.

[12] J. Spencer. Randomization, derandomization and antirandomization: Three games. *Theor. Comput. Sci.*, 131:415–429, 1994.