# Future of Voting

**Christian Reinbothe**
**Sudermanplatz 8 – 10**

**50670 Köln**
**Germany**

**mailto:Christian.Reinbothe@T-Online.DE**
**http://WWW.Reinbothe.DE**

# 1. Run-Off

We take a look at a normal democratically election. Assume that $n \in \mathbb{N}_+$ is the number of eligible voters and that we have two politians $P_1$ and $P_2$.

If the election is over, than there are $n_1, n_2 \in \mathbb{N}$ with

The politician $P_1$ got $n_1$ votes.
The politician $P_2$ got $n_2$ votes.
$n_1 + n_2 \leq n$

In this case we have:

Iff $n_1 > n_2$, $P_1$ has won the election.
Iff $n_1 < n_2$, $P_2$ has won the election.
Iff $n_1 = n_2$, there is a deadlock between $P_1$ and $P_2$.

Observation:

Iff the election is decided, then a lot of votes are lost.

## 2. Proportional Representation

We replace the two politicians $P_1$ and $P_2$ with political parties. Let $l \in \mathbb{N}_+$ be the chairs in the house of parliament. Then we have for $n_1 + n_2 > 0$:

$P_1$ got $\frac{n_1}{n_1 + n_2} \cdot l$ chairs.

$P_2$ got $\frac{n_2}{n_1 + n_2} \cdot l$ chairs.

This is much better than a run-off. Remarks:

A rounding error is possible.
If in doubt, the parties can make a compromise.
(So the parties can handle a deadlock.)

# 3. Suggestion

Instead of voting with as single vote for one party and computing the chairs, just vote the factors of *l*.

Let $k \in \mathbb{N}_+$ the number of involved political parties and let $P_1, \dots, P_k$ be these parties. First we define a function $F : \mathbb{R}^k \to \mathbb{R}^k$:

$$\forall \alpha \in \mathbb{R}^k \left( \alpha \neq 0 \Rightarrow \left( \forall \kappa \in \{1, \dots, k\} \quad (F(\alpha))_\kappa := \frac{\alpha_\kappa}{\sum_{j=1}^{k} |\alpha_j|} \right) \right)$$

$$F(0) := 0$$

This function is a kind of barycenter. It holds:

$$\forall \alpha \in \mathbb{R}^k \left( \alpha \neq 0 \Rightarrow \left( \sum_{j=1}^{k} \left| (F(\alpha))_j \right| = 1 \right) \right)$$

Cave!: You must understand the definition of $F : \mathbb{R}^k \to \mathbb{R}^k$. It computes the proportionally percentage of $|x_k|$ of $\sum_{j=1}^{k} |x_j|$. It normalizes $(x_1, \dots, x_k)$.

We define a second function $G : \mathbb{R}^k \to \mathbb{R}^k$

$$\forall \alpha \in \mathbb{R}^k \ \forall j \in \{1, \dots, k\} \quad \left( \left( \alpha_j \leq 0 \right) \Rightarrow \left( \left( (G(\alpha))_j \right) := 0 \right) \right)$$

$$\forall \alpha \in \mathbb{R}^k \ \forall j \in \{1, \dots, k\} \quad \left( \left( \alpha_j \geq 0 \right) \Rightarrow \left( \left( (G(\alpha))_j \right) := \alpha_j \right) \right)$$

This function forgets the parties with voting < 0!

Assume that $\left(v_1, \ldots, v_n\right) \in \left(\mathbb{R}^k\right)^n$ is the votings of the eligible voters. Then we have:

$F\left(v_i\right)$ is the wish of the voter $i$ for the factors of $l$.

Now we compute an intermediate result $\lambda \in \mathbb{R}^k$ as

$$\forall j \in \{1, \ldots, k\} \quad \lambda_j := \sum_{i=1}^{n} \left(F\left(v_i\right)\right)_j$$

(Vote Counting!)
(To count the votings all votes must be normalized.)

Finally, we can define the result $\Lambda \in \mathbb{R}^k$ of the election as

$$\Lambda := (F \circ G)(\lambda)$$

So every party $P_j$ gets $\left(\Lambda_j \cdot l\right)$ chairs in the house of parliament. Remarks:

A rounding error is possible.
The new algorithm is difficult.
The new algorithm needs an application to count the votes.
Negative voting is possible.

# 4. An Example Voting

You can do a ranking for the parties. Just give the parties a grading from 0 – 15:

“0”  = none approvement
…
“15” = most approvement

These are the school gradings in Germany.

# 5. Negative Voting

Consider the party $P_j$. If you give this party a “-1” and all other parties a “0”, you can compensate votings from other eli-gible voters a little bit.