Akademie věd České republiky
Ústav teorie informace a automatizace

Academy of Sciences of the Czech Republic
Institute of Information Theory and Automation

RESEARCH REPORT

Jiří Grim
Institute of Information Theory and Automation
Academy of Sciences of the Czech Republic

UNSUPERVISED VERIFICATION OF FAKE NEWS BY PUBLIC OPINION

No. 2390 June 2021

ÚTIA AV ČR, P.O. Box 18, 182 08 Prague, Czech Republic
E-mail: grim@utia.cas.cz
http://www.utia.cas.cz/people/grim

Unsupervised Verification of Fake News by Public Opinion

Jiří Grim
Institute of Information Theory and Automation of the Czech Academy of Sciences, Prague, Czech Republic

1 Introduction

In the last years we can see an extreme increase of communication possibilities. The most relevant feature of the new technical means like internet and social networks is the world-wide individual access to the communication space. Nearly everybody may present messages, news, opinions, artworks or other materials to the public networks in a way that makes them immediately accessible to a large community of users. Nonetheless, this new wonderful communication freedom is often disturbed by the bad quality and controversial background of the published content. There is an obvious need to identify fake news and to remove xenophobe, illegal or aggressive materials. Unfortunately, the problem is not solvable without limitations of the communication freedom and for this purpose we are missing legal authorities with desirable rights and responsibility.

In this paper we discuss a simple way to evaluate the messages in social networks automatically, without any special content analysis or external intervention. We presume, that a large number of social network participants is capable of a relatively reliable evaluation of materials presented in the network. Considering a simple binary evaluation scheme (like/dislike), we propose a transparent algorithm with the aim to increase the voting power of reliable network members by means of weights. The algorithm supports the votes which correlate with the more reliable weighted majority and, in turn, the modified weights improve the quality of the weighted majority voting. In this sense the weighting is controlled only by a general coincidence of voting members while the specific content of messages is unimportant. The iterative optimization procedure is unsupervised and does not require any external intervention with only one exception, as discussed in Sec. 5.2 .

In simulation experiments the algorithm nearly exactly identifies the reliable members by means of weights. Using the reinforced weights we can compute for a new message the weighted sum of votes as a quantitative measure of its positive or negative nature. In this way any fake news can be recognized as negative and indicated as controversial. The accuracy of the resulting weighted decision making was essentially higher than a simple majority voting and has been considerably robust with respect to possible external manipulations.

There is an extensive literature concerning weighted decision making. Most of the references relate to s.c. crowdsourcing systems (cf. e.g. [3]) or to learning with multiple labels (cf. e.g. [4]). The crowdsourcing systems usually assume a fixed set of simple tasks electronically distributed to numerous differently reliable piece-workers. The answers of workers are weighted according to their credibility with the aim to get the best possible solutions of the tasks [5], [3]. In case of multiple (noisy) labels the system evaluates the reliability of different annotators in order to get the best estimates of the actual hidden labels [6], [4]. There are different modifications of the above problems [1] but, unlike the weighted vote, the solution of underlying tasks is always of essential meaning and the accuracy of resulting answers or estimated labels is the primary goal.

The main motivation of the proposed algorithm is its application in a large social network. The content of evaluated messages is unimportant, only the related decision making of participants is registered and compared with the weighted vote with the aim to identify the most reliable voters. A large number of participants and communicated messages should enable to design a reliable and robust weighted voting scheme. Ideally the resulting weighted vote should provide a generally acceptable emotional feedback for network participants and could be used to indicate positive or controversial news in a suitably chosen quantitative way. The optimization algorithm has to be simple, transparent and intuitive to make the weighted vote well acceptable as a general evaluation tool.

2 Evaluation of Messages by Majority Voting

We consider a social network of arbitrary technical background unifying a large number of participants $x \in \mathcal{X}$ and offering very general communication possibilities. In particular, the members of social network may present arbitrary objects $a \in \mathcal{A}$ in the network being visible to other members. We assume various types of communicated content like assertion, information, news or messages and the members are supposed to have the possibility of a simple binary evaluation of the presented objects. Formally, the objects can be evaluated by the network members as positive or negative in a general sense.

We denote $x(a) \in \{-1, +1\}$ the binary evaluation of a message $a \in \mathcal{A}$ by the network member $x \in \mathcal{X}$, briefly a vote, whereby $x(a) = 1$ stands for like, agree, support and $x(a) = -1$ is understood e.g. as dislike, disagree or refuse. Simultaneously, we distinguish two basic types of messages, positive or negative, in accordance with the assumed binary voting. If we denote $\xi(a) \in \{-1, +1\}$ the unknown (hidden) type of the message $a \in \mathcal{A}$, then the correct evaluation of the message implies the equality $x(a) = \xi(a)$ or, equivalently, $x(a)\xi(a) = 1$. In case of incorrect evaluation we have $x(a) = -\xi(a)$ or, equivalently, $x(a)\xi(a) = -1$.

If we assume the knowledge of the true types $\xi(a)$ of the messages in the simulation experiments, then we can compute the mean decision accuracy (correct evaluation rate) p_x of the network members by Eq.

$$p_x = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{[\xi(a)x(a) + 1]}{2}, \quad 0 \leq p_x \leq 1, \quad x \in \mathcal{X}. \tag{1}$$

For simplicity we assume that all members $x \in \mathcal{X}$ evaluate all objects $a \in \mathcal{A}$, the number of elements in the sets will be denoted by $|\mathcal{A}|, |\mathcal{X}|$. In a practical situation the simplifying assumptions are not necessary, as discussed in Sec. 6.

Asymptotically, for a large number of messages, the value p_x can be interpreted as the probability of correct decision of the member $x \in \mathcal{X}$ and Eq. (1) can be rewritten in the form:

$$\frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)x(a) = p_x - (1 - p_x) = 2p_x - 1, \quad x \in \mathcal{X}. \tag{2}$$

Considering all network members $x \in \mathcal{X}$ we can compute the corresponding more reliable mean vote:

$$s(a) = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} x(a), \qquad -1 \leq s(a) \leq 1, \quad a \in \mathcal{A}. \tag{3}$$

The mean vote $s(a)$ naturally implies a simple majority decision $d(a)$ of the network members:

$$s(a) \geq 0 \ \Rightarrow d(a) = 1, \qquad s(a) < 0 \ \Rightarrow d(a) = -1, \qquad \Rightarrow \ d(a) = sign(s(a)), \quad a \in \mathcal{A}.$$

In analogy with (1) we can compute the mean accuracy of majority voting (correct decision rate of the simple majority vote):

$$\alpha = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{[\xi(a)d(a) + 1]}{2}, \quad 0 \leq \alpha \leq 1. \tag{4}$$

The mean "gain" of majority voting, defined by Eq.

$$\bar{s} = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)s(a), \tag{5}$$

can be interpreted as the mean difference between the correct ($\xi(a)x(a) = 1$) and incorrect ($\xi(a)x(a) = -1$) votes and refers to the reliability of majority voting. For a large number of messages $a \in \mathcal{A}$ the value of mean gain is related to the mean reliability of network members. Making substitution (3):

$$\bar{s} = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a) \left[\frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} x(a) \right] = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} \left[\frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)x(a) \right], \tag{6}$$

we can write by Eq. (2):

$$\bar{s} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} [2p_x - 1] = 2 \left[\frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} p_x \right] - 1 = 2\bar{p} - 1, \quad \bar{p} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} p_x. \tag{7}$$

3 Weighted Evaluation of Messages

The main idea of the paper is to differentiate the influence of network members according to the reliability of their votes. For this purpose we first introduce fixed voting weights v_x with the aim to increase the influence of reliable voters. Intuitively the fixed voting weight should be proportional to the probability of correct decision p_x of the member $x \in \mathcal{X}$:

$$v_x \approx p_x, \ \Rightarrow \quad v_x = \frac{p_x}{\sum_{x \in \mathcal{X}} p_x}, \ 0 \leq v_x \leq 1, \ (\sum_{x \in \mathcal{X}} v_x = 1), \quad x \in \mathcal{X}. \tag{8}$$

By means of the weights v_x we can compute the weighted evaluation of a message $a \in \mathcal{A}$ by the network members $x \in \mathcal{X}$, in analogy with the mean vote of Eq. (3):

$$\sigma_v(a) = \sum_{x \in \mathcal{X}} v_x x(a), \quad -1 \leq \sigma_v(a) \leq 1, \quad a \in \mathcal{A}, \tag{9}$$

and by using the weighted sum $\sigma_v(a)$ we define the corresponding weighted majority decision (weighted vote) $\delta_v(a)$:

$$\sigma_v(a) \geq 0 \ \Rightarrow \delta_v(a) = 1, \qquad \sigma_v(a) < 0 \ \Rightarrow \delta_v(a) = -1, \quad \delta_v(a) = sign(\sigma_v(a)). \tag{10}$$

It can be seen, that the resulting weighted vote $\delta_v(a)$ is more strongly influenced by the reliable members playing a dominant role in the weighted sum $\sigma_v(a)$. By Eq. (1) we can compute the accuracy of weighted voting (correct weighted vote rate):

$$\alpha_v = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{[\xi(a)\delta_v(a) + 1]}{2}. \tag{11}$$

In analogy with Eq. (5) the mean "gain" of weighted voting is defined by Eq. :

$$q_v = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)\sigma_v(a), \tag{12}$$

and can be interpreted as a mean difference between the correct and incorrect weighted vote. Again, the mean gain q_v is related to the weighted mean reliability of members:

$$q_v = \sum_{x \in \mathcal{X}} v_x [\frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)x(a)] = \sum_{x \in \mathcal{X}} v_x \left[2p_x - 1\right] = 2\left[\sum_{x \in \mathcal{X}} v_x p_x\right] - 1 = 2\bar{p}_v - 1, \tag{13}$$

defined by

$$\bar{p}_v = \sum_{x \in \mathcal{X}} v_x p_x. \tag{14}$$

4 Unsupervised Estimation of Voting Weights

In real-life situations, like e.g. in social networks, the true nature of messages is unknown. Consequently, the decision accuracy p_x is not available and has to be replaced in the definition of voting weights v_x by some other information. In order to optimize the "unsupervised" voting weights $\tilde{w}_x$ we construct a weighted sum $\sigma_w(a)$ analogous to $\sigma_v(a)$:

$$\sigma_w(a) = \sum_{x \in \mathcal{X}} \tilde{w}_x x(a), \quad -1 \leq \sigma_w(a) \leq 1, \quad a \in \mathcal{A}. \tag{15}$$

By using the weighted sum $\sigma_w(a)$ we can define the corresponding weighted majority decision (weighted vote) $\delta_w(a)$ in analogy to (10):

$$\sigma_w(a) \geq 0 \;\Rightarrow \delta_w(a) = 1, \quad \sigma_w(a) < 0 \;\Rightarrow \delta_w(a) = -1, \;\; \delta_w(a) = sign(\sigma_w(a)), \tag{16}$$

and the corresponding accuracy α_w, cf. (11):

$$\alpha_w = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \frac{[\xi(a)\delta_w(a) + 1]}{2}. \tag{17}$$

According to the main idea of the present approach, the unsupervised voting weight w_x of a network member $x \in \mathcal{X}$ should reflect the long term conformity of his decision making with the weighted majority vote. In this sense we define the weight w_x as a mean product of the vote $x(a)$ with the weighted sum $\sigma_w(a)$:

$$w_x = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} x(a)\sigma_w(a), \quad -1 \leq w_x \leq 1, \quad x \in \mathcal{X}. \tag{18}$$

In Eq. (18) the contribution $x(a)\sigma_w(a)$ is positive if the vote $x(a)$ coincides with the weighted vote $\delta_w(a)$ and it is negative if they are different. Simultaneously the value of the weighted sum $\sigma_w(a)$ differentiates between weak or strong coincidence or difference.

The formula (18) naturally implies the possibility of a negative weight $w_x < 0$ if the vote $x(a)$ frequently differs from the weighted vote $\delta_w(a)$. As the weights w_x may be negative, we use the norming by the absolute values $|w_x|$:

$$\tilde{w}_x = \frac{w_x}{\sum_{x \in \mathcal{X}} |w_x|}, \; -1 \leq \tilde{w}_x \leq 1, \quad \sum_{x \in \mathcal{X}} |\tilde{w}_x| = 1, \quad x \in \mathcal{X}. \tag{19}$$

Let us remark that, by means of the negative weights w_x, we can utilize even reliably incorrect voters (”protest voters”) to improve the accuracy of weighted majority voting.

In simulation experiments, when the true types of messages are available, we can compute the mean gain of the weighted vote δ_w by Eq.

$$q_w = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)\sigma_w(a) = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)[\sum_{x \in \mathcal{X}} \tilde{w}_x x(a)], \tag{20}$$

$$q_w = \sum_{x \in \mathcal{X}} \tilde{w}_x [\frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} \xi(a)x(a)] = \sum_{x \in \mathcal{X}} \tilde{w}_x \, [2p_x - 1] \, . \tag{21}$$

The value of mean gain q_w can be interpreted as an additional criterion of the weighted vote reliability. Roughly speaking, a high positive value of q_w implies that the correct weighted voting was well justified in the mean. On the other hand, the mean gain q_w near zero suggests the occurrence of both correct and incorrect weighted votes. The last Eq. (21) also illustrates the effect of negative weights $\tilde{w}_x$. In case of a protest voter the probability p_x is less than 0.5 and the expression $(2p_x - 1)$ is negative but the negative weight $\tilde{w}_x$ makes the contribution $\tilde{w}_x(2p_x - 1)$ positive.

It can be seen, that the above equations are iterative. Given the weights $\tilde{w}_x,\ x \in \mathcal{X}$ we can compute the weighted sums $\sigma_w(a)$ for the set of messages $a \in \mathcal{A}$ and by Eqs. (18), (19), we obtain the new values of the weights $\tilde{w}_x$. *In this way the algorithm combines the decision abilities of network members to get a more reliable weighted vote δ_w and, in turn, the underlying weighted sum σ_w is used to reinforce the weights $\tilde{w}_x$ of reliably voting members. Roughly speaking, the procedure is similar to a spontaneously arising public opinion (crowd wisdom) based on multiple information exchange in a community.* We recall that the Eqs. (15), (18) do not use the knowledge $\xi(a)$ of the true type of messages and therefore the optimization is unsupervised, the specific content of messages need not be analyzed. Let us remark that the intuitive idea of the proposed unsupervised learning is similar to a theoretically justified solution based on EM algorithm (cf. [2], Eq. (26)).

5 Simulation Experiments

The main goal of the simulation experiments is to demonstrate the possibility of unsupervised identification of reliable participants in a network and to realize a reliable weighted voting scheme. For this purpose we first generate a set of network participants with randomly specified reliability. According to Eq. (1) we denote by p_x the hidden probability of correct evaluation of messages by the member $x \in \mathcal{X}$:

$$p_x = P\{x(a) = \xi(a)\} \quad a \in \mathcal{A}, \quad (0 \le p_x \le 1), \quad x \in \mathcal{X}. \tag{22}$$

In other words, p_x is the hidden reliability of the network member $x \in \mathcal{X}$, with the following meaning: $p_x = 1$ means reliably correct evaluation, $p_x = 0$ implies reliably incorrect evaluation ("protest" voting) and $p_x = 0.5$ corresponds to completely random evaluation (random voting).

During the simulation experiment we generate a sequence of messages $a \in \mathcal{A}$ of randomly specified type $\xi(a)$. Given a random number r, $(0 \le r \le 1)$, we define the type of the message $a \in \mathcal{A}$ as negative or positive with equal probability:

$$r \le 0.5 \;\Rightarrow\; \xi(a) = 1, \quad r > 0.5 \;\Rightarrow\; \xi(a) = -1, \quad a \in \mathcal{A}.$$

Nevertheless, the proportion of negative and positive messages is irrelevant because the algorithm reflects only the coincidence of individual votes with the weighted majority.

Given a message of the type $\xi(a)$ we have to simulate the evaluation of the message by network participants randomly, according to the hidden probabilities p_x. In particular, given a random number r, the vote $x(a)$ is generated as correct if $r \le p_x$, $(\Rightarrow\ x(a) = \xi(a))$, or incorrect if $r > p_x$, $(\Rightarrow\ x(a) = -\xi(a))$.

At the beginning of the experiment we only need to specify the initial voting weights, e.g. as uniform: $w_x = 1/|\mathcal{X}|,\ x \in \mathcal{X}$. The adaptation of w_x based on the correlation of $x(a)$ with the weighted sum of votes $\sigma_w(a)$ is performed sequentially after each new message $a \in \mathcal{A}$ by Eqs.

$$\sigma_w(a) = \sum_{x \in \mathcal{X}} \tilde{w}_x x(a), \quad a \in \mathcal{A}, \ \tilde{w}_x = \frac{w_x}{\sum_{x \in \mathcal{X}} |w_x|}, \tag{23}$$

$$w'_x = \frac{1}{|\mathcal{A}|} \sum_{a \in \mathcal{A}} x(a)\sigma_w(a), \quad x \in \mathcal{X}. \tag{24}$$

The sequential optimization is motivated mainly by practical viewpoints, since otherwise it would be necessary to store the voting history and, consequently, the complexity and storage requirements could become prohibitive for a large number of messages and participants (cf. Sec. 6).

5.1 Standard Community of Network Participants

In the first experiment we assume rather sceptically the reliability p_x of 1000 standard participants to be slightly below the mean ($p_x = 0.45$), with only small subset being greater ($p_x > 0.45$) or even smaller ($p_x < 0.45$). In particular, we set first $p_x = 0.45$ for all $x \in \mathcal{X}, (|\mathcal{X}| = 1000)$ and generate for about 30% of participants the increased probabilities p_x randomly according to random numbers $0 \leq r \leq 1$ by Eq. $p_x = 0.45 + 0.15r$. Similarly we generate the decreased probabilities by Eq. $p_x = 0.45 - 0.15r$ for about 15% of participants. The resulting mean reliability of voters is $\bar{p} = 0.46$ (cf. (7)). The probabilities p_x are displayed as black columns in Fig. 1 in descending order, for better insight. The dot line indicates the value $p_x = 0.5$.

The adaptation of weights is shown in Fig. 1 at five stages of the sequential iteration process. Convergence of the voting weights is illustrated by 100 participants (every tenth), number of evaluated messages is IT=5000. The black columns correspond to the hidden reliability of the participants and the green columns to the corresponding voting weights. After 5000 iterations the optimized weights are higher for more reliable voters ($p_x > 0.5$) and nearly zero for less reliable voters ($p_x \approx 0.5$). The negative weights identify the "protest" voters ($p_x < 0.5$). Let us note that, in view of the definition of the weighted sum (15), the "reliable" protest voters with negative weights help to improve the decision accuracy. After 5000 iterations the resulting weighted voting achieves the accuracy 99%, which is essentially higher than that of simple majority (0.7%). The low accuracy of simple majority voting corresponds to the related mean gain $\bar{s} = -0.0747$, cf. (5), (7).

5.2 Global Inversion of Voting Weights

The only principle of the proposed unsupervised optimization of voting weights is the conformity of the individual voting of participants with the arising weighted majority voting. The optimization procedure doesn't require any external intervention with only one exception: if at the beginning the weighted majority decisions are strange or incorrect, e.g. because of unsuitable initial weights, then the optimized weights will reinforce the incorrect weighted majority voting, up to the extreme form. There is no internal mechanism to recognize the incorrect optimization. The only way is the external check of the obvious malfunction and inversion of the optimization process by changing the sign of all voting weights. As a result the weighted sum σ_w will change the sign in the next steps and the change will be reinforced by the correctly continuing optimization process.

In the simulation experiment we have the possibility to recognize the incorrect optimization by the negative gain of weighted voting q_w, cf. (20). Fig. 2 illustrates the situation with the initial incorrect weighted voting caused by initial parameters. At the beginning the accuracy of the weighted vote is zero, the weights of more reliable voters are negative (cf. IT=200, IT=500) and the weights of "protest" voters are positive. The sign of all weights is changed at iteration IT=1000 by external intervention. After this change the mean accuracy of the weighted vote slowly increases up to 80% at IT=5000. The final weights again correctly identify the reliable voters. On the other hand, the low accuracy of the simple majority vote (about 0.7%) is not influenced by the changing weights.

5.3 Coordinated Attempt to Change Voting

In the third experiment we illustrate the stability of weighted voting from the first example (Fig. 1) in case of coordinated attempt of a group of participants to invert the voting scheme. For this purpose we assume that after 1500 iterations a fixed group of 600 participants changes the voting properties towards protest voters with the aim to "switch" the optimization process. Formally, we set the corresponding 600 hidden probabilities to $p_x = 0.1$ at iteration IT=1500. Despite this change the mean accuracy of weighted vote does not decrease since the algorithm quickly identifies the protest voters by negative weights (cf. IT=2000, IT=5000) and the resulting accuracy is about 99% again.

Another possibility would be a coordinated attempt of a group of voters to influence a single evaluation of a given message. Theoretically the attempt could succeed, however in case of a real network the optimized weighted vote is supposed to be based on a very large anonymous community of participants and therefore any realistic number of coordinated protest voters would be probably insufficient.

6 Practical Implementation of Weighted Voting

In a real life situation like social network we have to assume that only a small subset of participants evaluates a particular message. We denote $\mathcal{X}_a$ the set of participants who evaluated a message $a \in \mathcal{A}$ and $\mathcal{A}_x$ the set of messages evaluated by the participant $x \in \mathcal{X}$. In order to avoid less relevant weighted votes we should restrict ourselves only to sufficiently "popular" messages $a \in \mathcal{A}$ satisfying a reasonable condition $|\mathcal{X}_a| > N_0$, i.e. with the number of voters greater than a suitable threshold N_0. For the sake of evaluation of a given message $a \in \mathcal{A}$ we have to store for every voter $x \in \mathcal{X}_a$ the vote $x(a)$, its current weight w_x and the number of evaluated messages $|\mathcal{A}_x|$. At any time these data can be used to compute the weighted sum

$$\sigma_w(a) = \sum_{x \in \mathcal{X}_a} \tilde{w}_x x(a), \quad \tilde{w}_x = \frac{w_x}{\sum_{x \in \mathcal{X}_a} |w_x|}. \tag{25}$$

and the corresponding updates of weights (cf. (23)) for all participants $x \in \mathcal{X}_a$:

$$w'_x = \frac{1}{(|\mathcal{A}'_x|)} \sum_{a \in \mathcal{A}'_x} x(a)\sigma_w(a) = \frac{1}{(|\mathcal{A}_x| + 1)}(|\mathcal{A}_x| w_x + x(a)\sigma_w(a)), \quad x \in \mathcal{X}_a. \tag{26}$$

Obviously the evaluation of multiple messages can proceed sequentially, in parallel, for a number of messages and stopped when the number of voters $|\mathcal{X}_a|$ is large enough. Repeated evaluation of messages is questionable because of possible noisy feedback.

Figure 4 shows the changing weights in case "realistic" simulation. In comparison with the Sec. 5.1 the only difference is the number of voting participants. Every voter is included into evaluation of a particular message randomly, with the probability 0.5. There are no apparent differences in both experiments since the number of randomly chosen voters (about 500) is obviously sufficient to guarantee a reasonable convergence.

7 Conclusion

Weighted voting should never replace the standard democratic voting principle based on equal votes. However, in some situations, the weighted voting could facilitate quick and valuable feedback between official institutions and public opinion. For example, there is a great area of popular cultural activities undesirably dominated by financial profit. For good reasons any censorship is not applicable in these cases but a well functioning weighted voting system could be applied to recognize and support high quality cultural products, e.g. by decreased taxes.

Acknowledgment

Thanks are due to the colleagues Petr Somol and Stefan Kroemer for valuable discussions.

References

[1] Dawid, A.P. and Skene, A.M. : Maximum likelihood estimation of observer error-rates using the em algorithm. Journal of the Royal Statistical Society. Series C (AppliedStatistics), 28 (1):20 - 28,1979.

[2] Grim J. : Individualized voting scheme - a democratic violation of the democratic voting principle, *Cybernetics and Systems '94*, p. 1081 - 1088, Eds: Trappl R., World Scientific, (Singapore 1994), European Meeting on Cybernetics and Systems Research /12./, (Vienna, AT, 05.04.1994 - 08.04.1994) [1994]

[3] Raykar, V.C., Yu, S., Zhao, L.H., Valadez, G.H., Florin, C., Bogoni, L. and Moy L.: Learning from crowds. J. Mach. Learn. Res., 99:12971322, August 2010.

[4] Sheng, V.S., Provost, F. and Ipeirotis, P.G.: Get another label? Improving data quality and data mining using multiple, noisy labelers. In Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD 08, pages 614 - 622. ACM, 2008.

[5] Slevin, D.P., Boone, L.W., Russo, E.M. and Allen, R.S. : CONFIDE: A collective decision-making procedure using confidence estimates of individual judgements, *Group Decision and Negotiation*, Vol. 7, No. 2, pp. 179–194, Springer [1998]

[6] Whitehill, J., Ruvolo, P., Wu, T., Bergsma, J., and Movellan, J.: Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. Advances in Neural Information Processing Systems, 22:2035 - 2043, 2009.

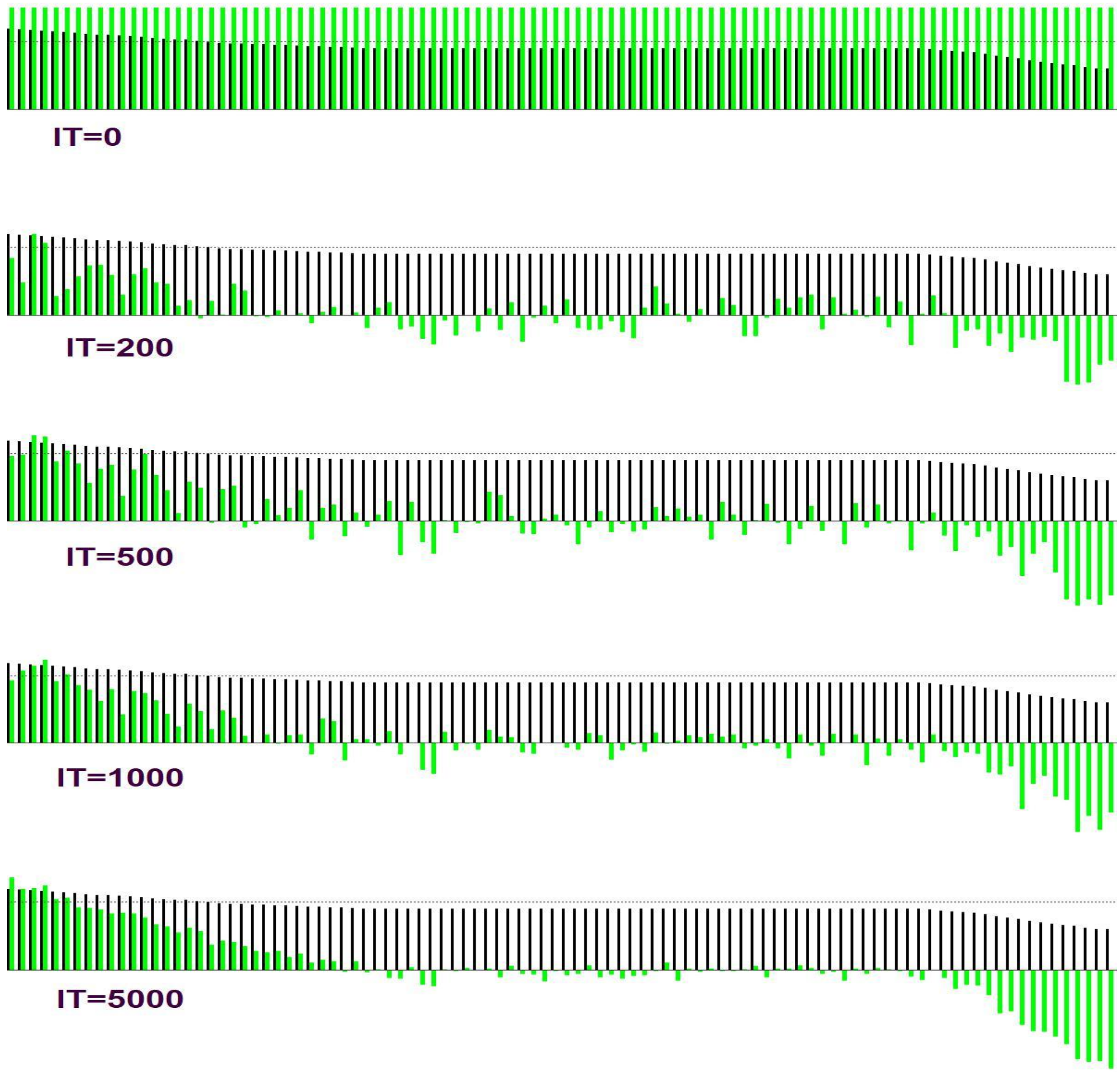

Figure 1: Convergence of the voting weights for 1000 members (only every tenth member is shown) number of evaluated messages is IT=5000. The black columns correspond to the hidden reliability of voters ($0 < p_x < 1$), the dot line indicates the value $p_x = 0.5$. The green columns correspond to the estimated weights. The probabilities p_x are generated randomly but they are ordered in descending way to make the relation to changing weights more apparent. The resulting weights are higher for more reliable voters ($p_x > 0.5$) and nearly zero for less reliable voters ($p_x \approx 0.5$). The negative weights identify the "protest" voters ($p_x \ll 0.5$). We recall that, in view of Eq. (15), the "reliable" protest voters actually help to improve the decision quality. After 5000 iterations the resulting weighted voting achieves the accuracy 99%, which is essentially higher than that of simple majority (0.7%).

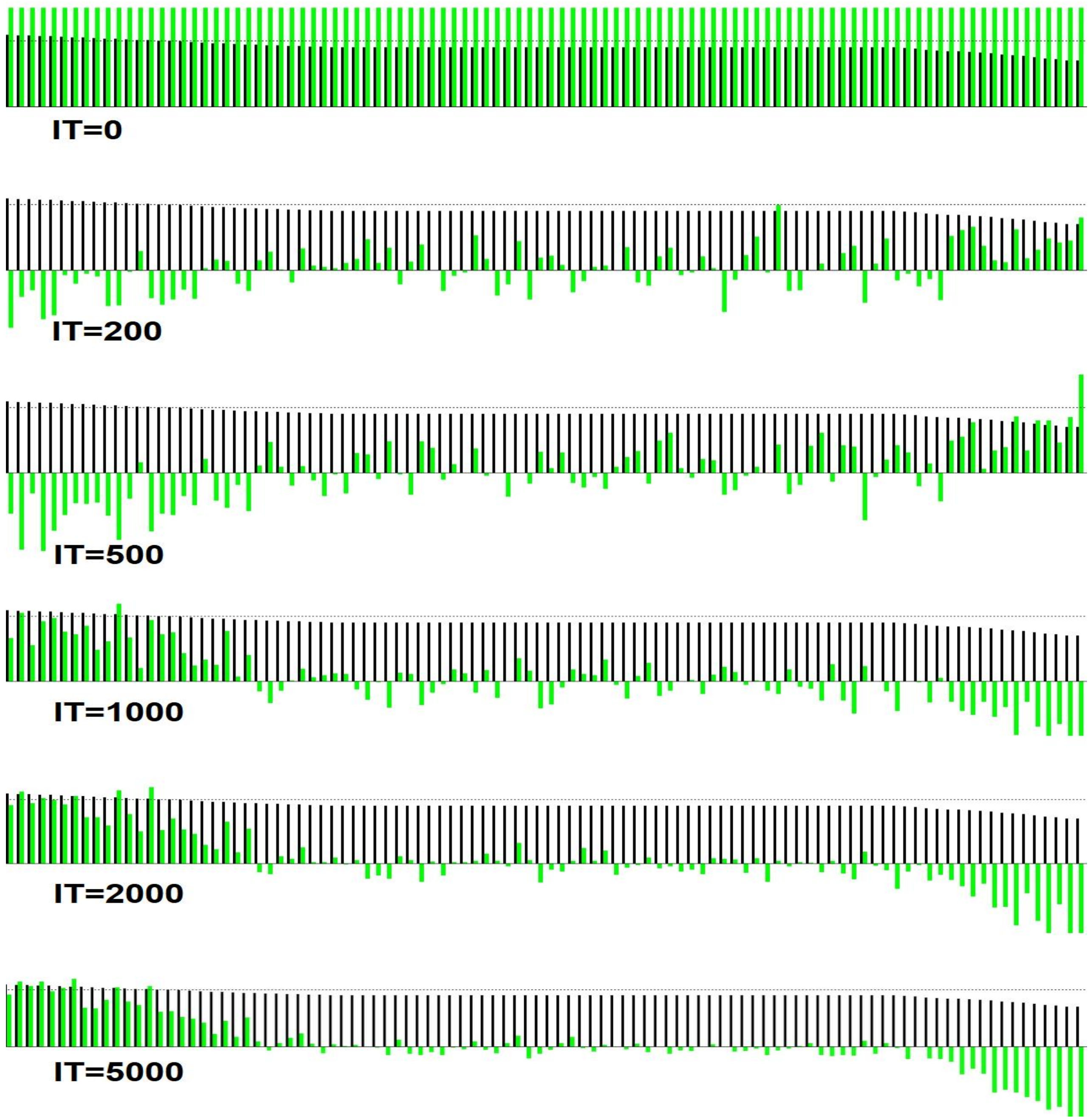

Figure 2: The Figure illustrates the situation with the initial incorrect weighted voting. The incorrect weights may arise if we change the initial parameters. In this figure the reliability of participants from the first experiment is slightly decreased. At the beginning the accuracy of the weighted vote is zero and the weights of more reliable voters tend to be negative. The sign of all weights is changed at iteration IT=1000 by external intervention. After this change the accuracy of the weighted vote slowly increases up to 80% at IT=5000. The final weights again correctly identify the reliable voters or protest voters. On the other hand, the accuracy of the simple majority vote corresponds to the decreased reliability of voters without any possibility to improve spontaneously.

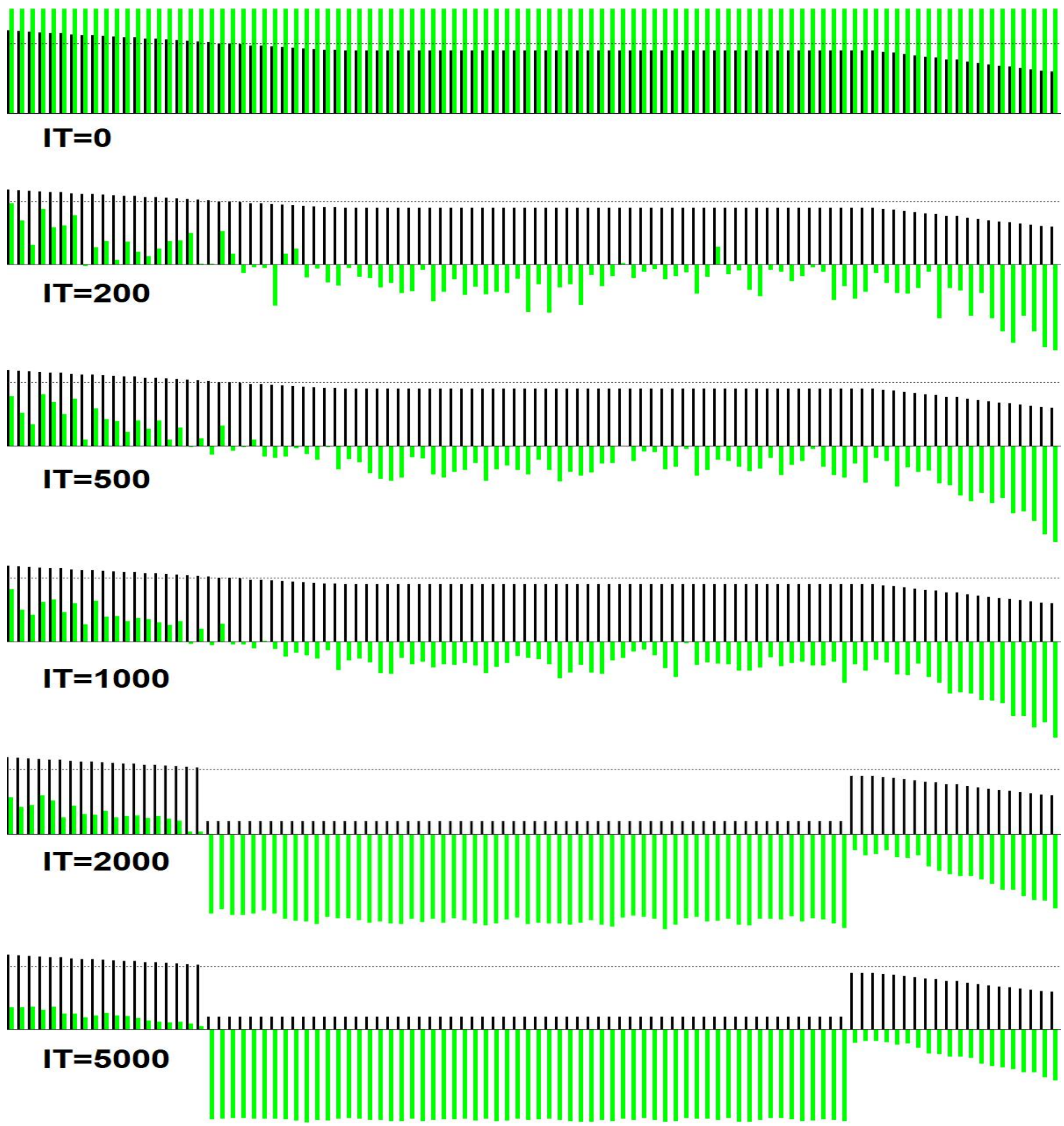

Figure 3: In the third experiment we illustrate the stability of weighted voting from the first example (Fig. 1) in case of coordinated attempt of a group of participants to change the resulting vote. For this purpose we assume that after 1500 iterations a fixed group of 60% of participants changes the voting properties towards protest voters. Formally, we set the corresponding hidden probabilities to $p_x = 0.1$. The accuracy of weighted vote does not decrease, as it can be seen, the algorithm quickly identifies the protest voters by negative weights and the resulting accuracy 99.88% is even higher. On the other hand, the accuracy of the simple majority vote starts at the value about 2%, and after the coordinated intervention decreases to 0.2%.

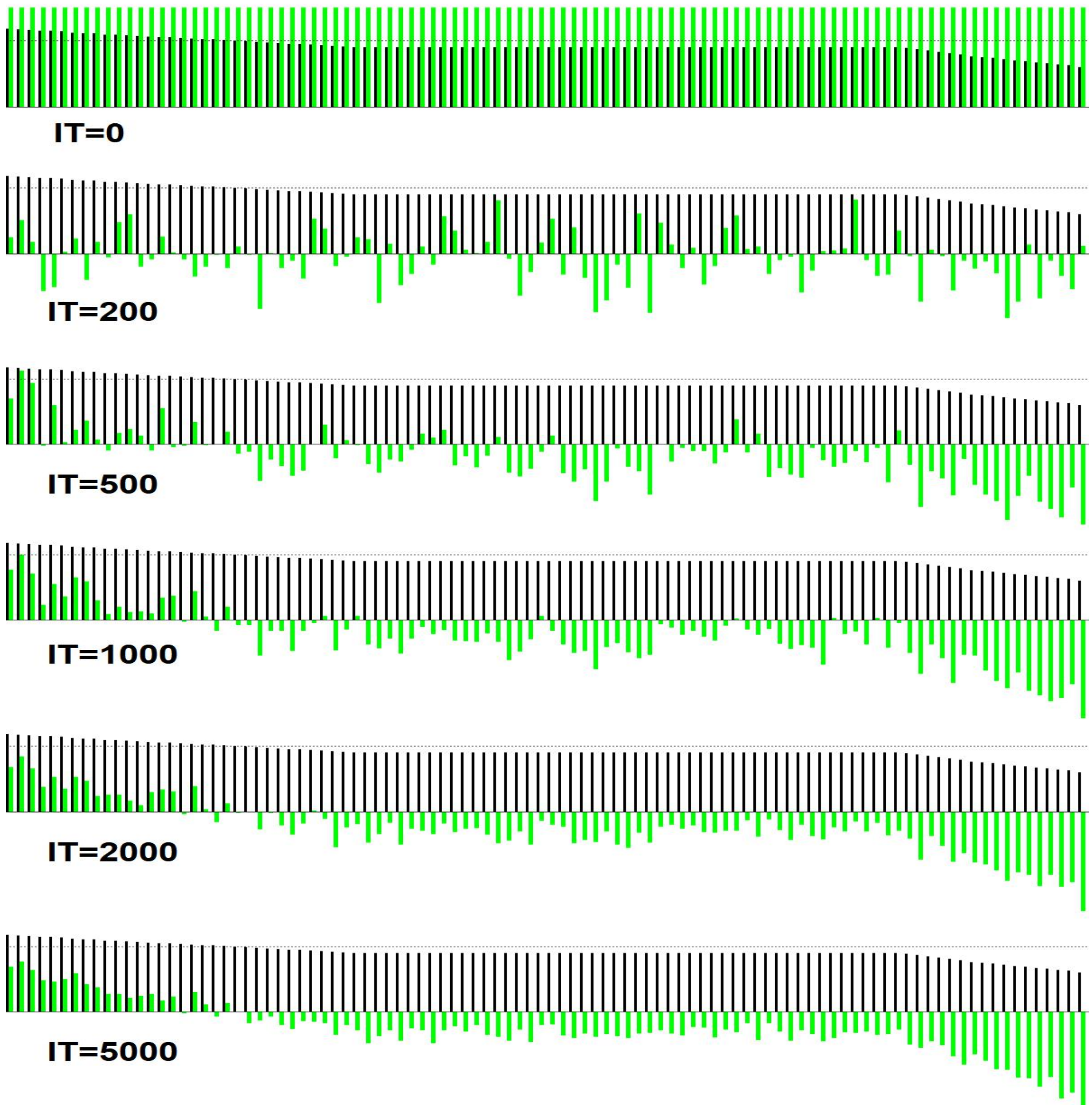

Figure 4: The fourth experiment shows the changing weights in case "realistic" simulation (cf. Sec. 6). In comparison with the Sec. 5.1 the only difference is the number of voting participants. Every voter is included into evaluation of a particular message randomly, with the probability 0.5, i.e. the number of voters is about 500. There are no apparent differences in both experiments since the number of voters is obviously sufficient to guarantee a reasonable convergence.