Mathematical Formalism
for Voting Process in Bitcoin

BitFury Group

Sep 10, 2015

Abstract

We propose a mathematical formalism for the voting process in Bitcoin ecosystem. This formalism can be used to algorithmically determine the best value of a certain parameter (e.g., a block size limit) which will be considered appropriate by all voters. The proposed approach is aimed to clarify vagueness of some proposed voting processes that potentially allows a party with marginal voting power to dictate their conditions to the rest of the network. To solve the problem, we introduce a non-negative dissatisfaction function and minimize its value summed over all votes. The value of the block size limit (and, potentially, other parameters of the protocol) found this way will satisfy voters provided the dissatisfaction function is chosen appropriately.

Version History

Version	Date	Change description
1.0	Sep 10, 2015	Initial version

Public voting is one of mechanisms to reach consensus in a decentralized environment such as Bitcoin. Bitcoin already has seen voting among miners implemented in BIP 34, which introduced a block height into coinbase scripts [1]. One of the current proposals to increase block size limit – BIP 100 – uses a similar approach [2, 3]. BIP 100 introduces a floating limit on block size, which is decided in voting rounds among miners. To vote, a miner must include a value of the limit he agrees with into the coinbase scripts. Voting rounds span 12,000 blocks (approximately 3 months). Allowed votes range between 80% and 120% of the current block size limit.

The weak point of BIP 100 is the mechanism how votes are tallied up. It uses the 20th and the 80th percentiles of the votes [3]:

- if the 20th percentile vote is higher than the current block size limit, the block size limit is increased to this vote

- if the 80th percentile vote is lower than the current block size limit, the block size limit is increased to this vote

- otherwise, the block size limit is unchanged.

In this case, a miner or a group of miners with slightly over 20% hash rate can hold back all attempts to increase block size limit (e.g., they can do so because they use outdated equipment that won't run with bigger block size).

There are other less formalized proposals to use voting among certain groups of users to determine block size limit. We propose a generic approach where the target block size limit is derived from the votes with relatively simple equations using optimization theory. Namely, we introduce a non-negative dissatisfaction function and minimize its value summed over all votes. The block size limit found this way would satisfy voters provided the dissatisfaction function is chosen appropriately. The model is not susceptible to 21% attack provided adequate parameters are chosen for it. The proposed approach with minimal changes could be used to vote other changes to the Bitcoin protocol. Our enhancement should not be considered final; it needs to be verified by the Bitcoin community and modeled to highlight possible issues.

Problem. We have block size limit votes $v_i > 0$, $i = 1, 2, \ldots, M$. (In BIP 100, $M = 12000$.) We need to determine the block size limit s that satisfies votes in a best way.

We can consider values v_i as a ratio of an actual size to the current block size limit. Alternatively, we can use votes measured in a certain unit (e.g., megabytes in case of the block size limit). In both cases, we assume votes are contained in a certain predefined interval (e.g., $0.5 \leqslant v_i \leqslant 2$).

Solution. Consider a function of two variables $F : (0, +\infty)^2 \to [0, +\infty)$. The value $F(s, v)$ expresses dissatisfaction of a user that voted for block size limit v, if the vote will result in limit s. Obviously, $F(v, v) = 0 \quad \forall v$. The problem is therefore reduced to an optimization problem

$$\hat{s} = \arg\min_s \sum_{i=1}^{M} F(s, v_i). \tag{1}$$

If we group votes for the same limit into blocks $v_{(1)} < v_{(2)} < \cdots < v_{(k)}$, with i-th block weight

$$w_i \in (0, 1), \quad \sum_{i=1}^{k} w_i = 1,$$

then (1) is equivalent to

$$\hat{s} = \arg\min_s \sum_{i=1}^{k} w_i F(s, v_{(i)}) \stackrel{\text{def}}{=} \arg\min_s L(s), \tag{2}$$

where $L(s)$ is the target function we need to minimize.

Note that (2) allows for the case in which the votes v_i are weighted themselves (e.g., if users vote with their Bitcoin funds).

Example. Consider the simplest case: a binary vote where there are only two possible outcomes; i.e., $v_i \in \{0, 1\}$ and $s \in \{0, 1\}$. Let $s = 1$ represent the vote outcome in which voters are required to perform some action (e.g., upgrade their equipment, install new software, etc.), and $s = 0$ represent the inaction outcome.

The dissatisfaction function is defined as

	$s = 0$	$s = 1$
$v = 0$	$F(0,0) = 0$	$F(1,0) = K$
$v = 1$	$F(0,1) = 1$	$F(1,1) = 0$

Typically, $K > 1$: a voter who does not want changes is inconvenienced by the action outcome $s = 1$ more than a person who voted for an action outcome if it was not implemented. This is especially true when the voting process is constant and there is a hope that the action outcome will be chosen in the observable future.

Assume there was p fraction of votes for action and $1 - p$ for inaction.

$$L(0) = pF(0,1) = p; \quad L(1) = (1 - p)F(1,0) = (1 - p)K.$$

The action outcome is preferable if

$$L(1) < L(0); \quad (1 - p)K < p; \quad p > \frac{K}{K + 1}.$$

BIP 34 implemented this kind of voting to determine a transition to version 2 blocks. The transition was considered accomplished once 95% of recent blocks contained version 2 in their header. This corresponds to

$$\frac{K}{K + 1} = 0.95; \quad K = 19.$$

This example shows that our approach merely develops ideas already circulating in the Bitcoin community.

1 Dissatisfaction Functions

What kind of a dissatisfaction function F should we consider? Consider two classes of functions that reduce F to a function of one argument:

- $F(s, v) = D(s - v)$, i.e., voter's dissatisfaction depends on an *absolute* difference between the target size and and his vote. In this case, target function

$$L(s) = \sum_{i=1}^{k} w_i D(s - v_{(i)}). \tag{3}$$

- $F(s, v) = D(s/v - 1)$, i.e., voter's dissatisfaction depends on a *relative* difference between the target size and and his vote. In this case,

$$L(s) = \sum_{i=1}^{k} w_i D(s/v_{(i)} - 1). \tag{4}$$

Function D should satisfy

$$D(0) = 0; \quad \forall x \; D(x) \geqslant 0; \quad \lim_{x \to \pm\infty} D(x) = +\infty; \tag{5}$$

Additionally, we will consider continuously differentiable functions D in order to find the minimum by solving $L'(s) = 0$; we don't lose much generality by considering this class of functions.

1.1 Quadratic Function

We can consider

$$D_{sq}(x) = \begin{cases} x^2, & \text{if } x \geqslant 0, \\ \alpha x^2, & \text{if } x < 0; \end{cases}$$

Logically, $\alpha < 1$ (a voter does not like situation where a target limit s is bigger than his vote v, but is partially satisfied with the situation where $s < v$). A couple of D_{sq} functions are plotted on Figure 1.

Absolute difference: $F(s, v) = D_{sq}(s - v)$. If we assume $\alpha = 1$ in D_{sq}, the solution to (2) will be the weighted average for all votes:

$$\hat{s} = \sum_{i=1}^{k} w_i v_{(i)}.$$

In the general case, we should differentiate (2) for each interval $(v_{(i)}, v_{(i+1)})$, $1 \leqslant i < k$; plus test all margin points $v_{(i)}$. The optimal point on interval $(v_{(i)}, v_{(i+1)})$ can be found as

$$\frac{dL}{ds} = \sum_{j=1}^{i} 2w_j(s - v_{(j)}) + \sum_{j=i+1}^{k} 2\alpha w_j(s - v_j) = 0;$$

$$\hat{s}_i = \left(\sum_{j=1}^{i} w_j v_{(j)} + \alpha \sum_{j=i+1}^{k} w_j v_{(j)} \right) \bigg/ \left(\alpha + (1 - \alpha) \sum_{j=1}^{i} w_j \right). \tag{6}$$
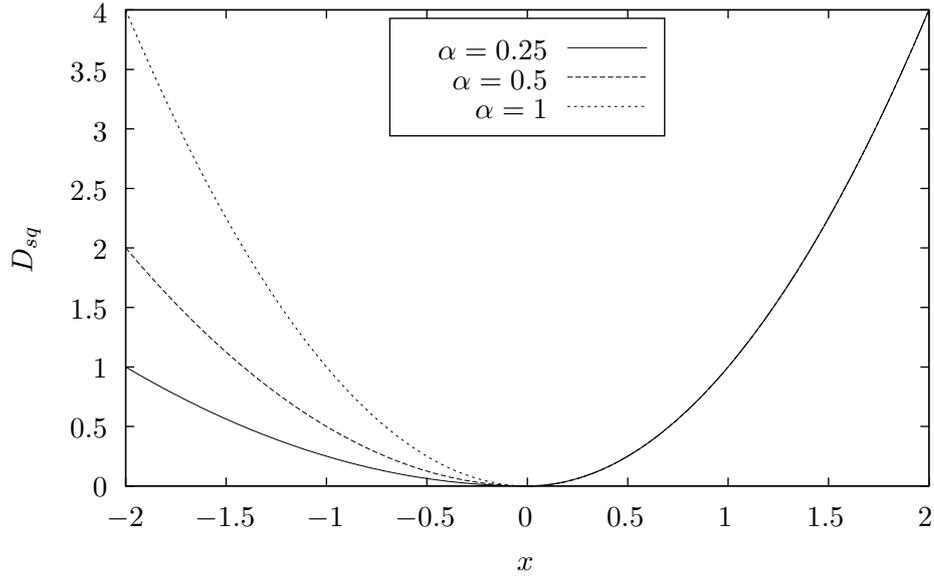
Figure 1: Function $D_{sq}(x)$ for various values of the parameter α.

The global optimum of the target function L

$$\hat{s} = \arg\min_{s \in S} L(s),$$

where a set of candidate points

$$S = \{\hat{s}_i\}_{i=1}^{k-1} \cup \{v_{(i)}\}_{i=1}^{k}.$$

Note that some $\hat{s}_i$ may not belong to the corresponding interval $(v_{(i)}, v_{(i+1)})$; we can exclude these points from the consideration. Furthermore, as the target function L is convex, we can skip checking marginal points $v_{(i)}$.

Relative difference: $F(s, v) = D_{sq}(s/v - 1)$. If we assume $\alpha = 1$ in D_{sq},

$$L(s) = \sum_{i=1}^{k} w_i(s/v_{(i)} - 1)^2;$$

$$\frac{dL}{ds} = \sum_{i=1}^{k} \frac{2w_i}{v_{(i)}}\left(\frac{s}{v_{(i)}} - 1\right) = 0;$$

$$\hat{s} = \sum_{i=1}^{k} \frac{w_i}{v_{(i)}} \Big/ \sum_{i=1}^{k} \frac{w_i}{v_{(i)}^2}.$$

For $\alpha \neq 1$ we can get an analytical solution by differentiating L on each of intervals $(v_{(i)}, v_{(i+1)})$ and checking margin points, just as in the previous case.

Example. Consider the following vote distribution for a block size limit:

i	1	2	3	4
Vote $v_{(i)}$, MB	1	2	4	8
Weight w_i	0.21	0.25	0.25	0.29

Examine the case (3) when dissatisfaction depends on the absolute difference between the vote and the target limit. First, when $\alpha = 1$, we get

$$\hat{s}(\alpha = 1) = 0.21 \cdot 1 + 0.25 \cdot 2 + 0.25 \cdot 4 + 0.29 \cdot 8 = 4.03.$$

Consider $\alpha = 0.5$. First, let's calculate target function at votes $v_{(i)}$. Vote for 4 is the optimal among all votes with total dissatisfaction $L(4) \approx 5.21$; however, the optimum of L does not generally coincide with one of the votes. We need to calculate the optimal size limit according to (6) on three intervals:

Interval	1 to 2	2 to 4	4 to 8
$\hat{s}_i$	3.504	3.247	3.357

Among these three values, only the second one fits into the corresponding interval. As $L(3.247) = 4.796 < L(4)$, we can conclude that 3.247 is indeed an optimal block size limit: $\hat{s}(\alpha = 0.5) = 3.247$. See Table 1 for more detailed calculations.

Table 1: Dissatisfaction of voters with possible vote outcomes for the quadratic function D_{sq} with parameter $\alpha = 0.5$. Total dissatisfaction is calculated as a weighted sum of voters' dissatisfactions

	Weight	Vote $v_{(i)}$,	Candidate limit s, MB				Result, MB
	w_i	MB	1	2	4	8	3.247
Voter's	21%	1	0	1	9	49	5.05
dissatisfaction	25%	2	0.5	0	4	36	1.56
	25%	4	4.5	2	0	16	0.28
	29%	8	24.5	18	8	0	11.3
Total dissatisfaction $L(s)$			8.36	5.93	5.21	23.3	**4.80**

If we take $\alpha = 0.25$, we will obtain $\hat{s}(\alpha = 0.25) = 2.588$; $\hat{s}(\alpha = 0.125) = 2.133$. Thus, the lower the value of α, the lower the optimal block size limit.

1.2 Exponential function

Consider a parameterized function

$$D_{exp}(x) = e^{\alpha x} + \alpha e^{-x} - \alpha - 1.$$

One can see that D_{exp} satisfies (5). $\alpha > 1$ corresponds to the case when a voter prefers a target limit less than his vote rather than an opposite situation. See Figure 2 for examples.

The derivative
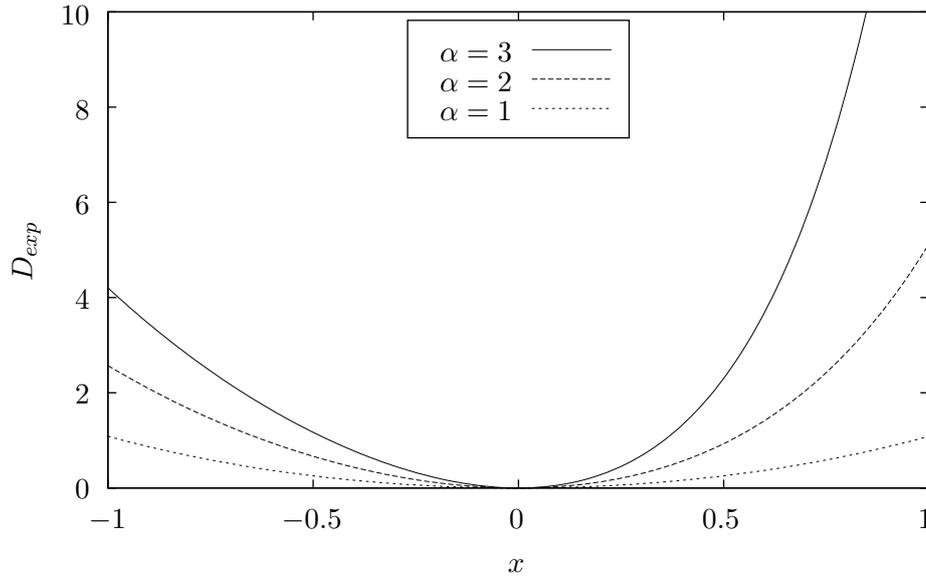
$$D'_{exp}(x) = \alpha(e^{\alpha x} - e^{-x}).$$

Figure 2: Function $D_{exp}(x)$ for various values of the parameter α.

Absolute difference: $F(s, v) = D_{exp}(s - v)$.

$$L'(s) = \sum_{i=1}^{k} w_i D'_{exp}(s - v_{(i)}) = 0;$$

$$\sum_{i=1}^{k} w_i \alpha \left[e^{\alpha(s - v_{(i)})} - e^{v_{(i)} - s} \right] = 0.$$

From here, we calculate the optimal block limit

$$\hat{s} = \frac{1}{\alpha + 1} \left(\ln \sum_{i=1}^{k} w_i e^{v_{(i)}} - \ln \sum_{i=1}^{k} w_i e^{-\alpha v_{(i)}} \right) \tag{7}$$

Relative difference: $F(s, v) = D_{exp}(s/v - 1)$. An explicit analytic expression for the optimal block limit is impossible in this case; one has to use computational methods to find it. As D_{exp} and consequently L are convex, calculating the optimal size limit is simple: one just has to find a single root of $L'(s) = 0$, e.g. by a binary search on $[v_{(1)}, v_{(k)}]$.

Example. Consider the same vote distribution as in the previous example. Examine the case (3) when dissatisfaction depends on the absolute difference between the vote and the target limit. Using (7), we get the following results:

α	1	2	3	4	5
Optimal block size $\hat{s}(\alpha)$	4.47	3.40	2.82	2.46	2.22

One can see that the parameter α indeed can be used to regulate the results of the vote: the higher the value of α, the lower the optimal block size limit. Note that when $\alpha = 1$, we get a result close to the weighted average of votes (which is equal to 4.03).

2 Further Research

One could introduce other functions F or D that may reflect dissatisfaction better than quadratic and exponential functions we have explored. Additionally, modeling needs to be performed in order to find optimal parameters of dissatisfaction functions (α in case of D_{sq} and D_{exp}).

References

[1] *Gavin Andresen* (2012). Block v2, height in coinbase (BIP 34)
    URL: `https://github.com/bitcoin/bips/blob/master/bip-0034.mediawiki`

[2] *Jeff Garzik* (2015). Making decentralized economic policy
    URL: `http://gtf.org/garzik/bitcoin/BIP100-blocksizechangeproposal.pdf`

[3] *Jeff Garzik* (2015). Floating block size hard limit (BIP 100)
    URL: `https://github.com/jgarzik/bip100/blob/master/bip-0100.mediawiki`