Discouragement Attacks

Vitalik Buterin
Ethereum Foundation

July 9, 2017

Abstract

We explore "discouragement attacks" on economic consensus mechanisms. A discouragement attack consists of an attacker acting maliciously inside a consensus mechanism in order to reduce other validators' revenue, even at some cost to themselves, in order to encourage the victims to drop out of the mechanism. The motivations to conduct discouragement attacks are twofold. First, the attacks can increase the attacker's profit, as the mechanism may contain long-run "competitive" mechanics where some validators dropping out increases revenue to the remaining ones. Second, the attacks can be part of a two-step strategy where the second step is to carry out a traditional 51% attack on the consensus algorithm against a now much smaller set of "honest" validators warding off the attacker, and hence pay a much lower cost for the attack.

1 Introduction

We model an economic consensus mechanism as being a game where there is an infinite set of validators each with an infinitesimally small deposit, with the total deposit size D, of which some portion is controlled by the attacker. The payout function takes as input D, the total deposit size, and h, the extent to which the attacker deviates from an "honest" strategy. The payout to each honest validator is $\frac{1-h}{D^p}$, where p is a protocol parameter that determines how the protocol reward changes with the number of validators. For example:

- $p = 0$: constant "interest rate", eg. under optimal conditions each validator earns a return of 8% per year.
- $p = \frac{1}{2}$: the rewards (and penalties) to validators scale with the inverse square root of the total deposit size, so *total* rewards scale with the square root of the total deposit size. This is a compromise between $p = 0$ and $p = 1$.
- $p = 1$: constant total reward, ie. the total payout of the protocol is dependent only on what percentage of validators take what actions, not on the total deposit size.
- $p = \infty$: the protocol is dead-set on ensuring that the total deposit size is some specific constant D_k no matter what. If the total deposit size exceeds D_k, the protocol keeps decreasing rewards until it drops to D_k, and if the total deposit size is below D_k, the protocol keeps increasing rewards until it rises to D_k.

Note that if revenues to validators are dominated by transaction fees, then $p = 1$ will hold.

Each validator controlled by the attacker gets a return of $\frac{1-\frac{h}{r}}{D^p}$ where r is the *proportional loss ratio*. The proportional loss ratio is the ratio between the loss the victims suffer and the loss the attacker suffers, where both losses are expressed in percentage terms. For example, if an attack that causes the attacker to lose 1% of deposits of all validators that they control causes everyone else to lose 2%, then the proportional loss ratio is 2.

The reason behind the above formulas is as follows. We assume that there is some "base interest rate" paid to all validators, which is proportional to some inverse power of the total deposit size. This is certainly not an exhaustive characterization of ways to assign the base interest rate based on the total deposit size, but inverse powers are attractive because they are robust to uncertainty; that is, if one designs a protocol using such a function with the expectation that the total deposit size will usually be X, but then in the real world the total deposit size unexpectedly turns out to be $10 * X$, the economics do not substantively change. There is not necessary a principled in-protocol notion of the "extent" to which an attacker is attacking, so we define our own: the extent of an attack is h if the victims' return decreases to $\frac{1-h}{x^p}$. We assume the proportional loss ratio r is fixed, hence the attacking validators' return must be $\frac{1-\frac{h}{r}}{x^p}$.

In contracts the *griefing factor*, another way of comparing attacker and victim losses, is defined in absolute terms: for example, if in such a scenario the attacker controls $\frac{1}{3}$ of the total validator set, then the set of victims is twice as large as the attacker, and so altogether the victims lose four times more than the attacker, and so the griefing factor would be 4. The relationship between the proportional loss ratio r and griefing factor is simple:

$$g = r * \frac{1-\alpha}{\alpha},$$

where α is the portion of validators controlled by the attacker. In our above example, $\alpha = \frac{1}{3}$, so $g = 2 * \frac{\frac{2}{3}}{\frac{1}{3}} = 4$.

We now rephrase the problem into the language of supply and demand: there exist a set of players, each of which has some *reserve interest rate* at which they are willing to become validators in the consensus mechanism. This is the demand curve, where the interest rate is the price. The protocol, which offers interest rates for participation in the consensus mechanism, sets the supply curve. If $p = 0$, the supply curve is horizontal - the protocol offers that interest rate to an unlimited number of validators. If $p = \infty$, the supply curve is vertical. For any other p, the supply curve is declining with a constant elasticity of $\frac{1}{p}$. We model the attacker as having unilateral power to set h (by attacking), and this pushes down the supply curve.

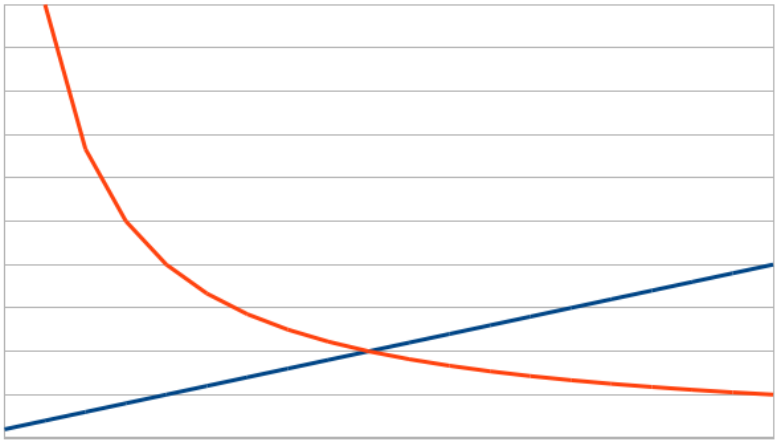
We model the demand curve as also being a simple exponential function, x^d. In general, we expect there to be wide disparities between the reserve interest rates of different players, as they have different levels of wealth, technical capability to operate a node in the consensus mechanism, and willingness to lock up their capital to become a validator; additionally, we expect many players will be readily willing to lock up 50% of their capital, somewhat willing to lock up 80%, hard pressed to lock up 95%, and not willing at all to lock up 100%. Hence, $d > 1$ seems likely, though we will consider the problem abstractly and give results for various values of d.

2 Analysis

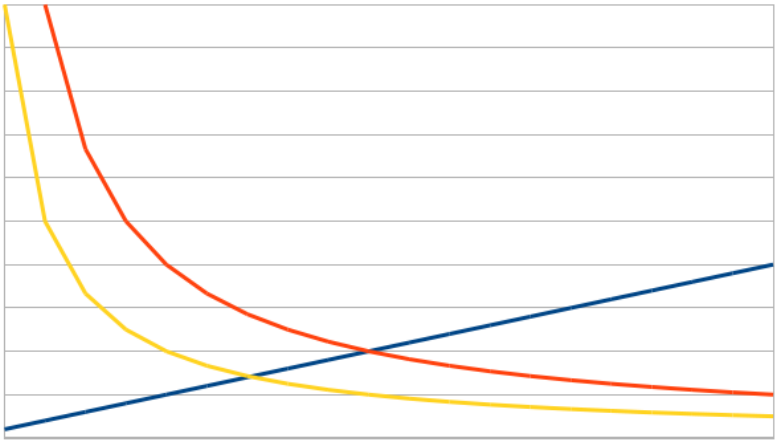
We want to learn two things. First, are there opportunities to perform a discouragement attack for profit? Second, what is the difficulty of performing a discouragement attack in order to set up a cheaper later attack on consen-

sus? To examine the second case, we can compare the pre-discouragement and post-discouragement intersections of the supply and demand curves.

Pre-discouragement, the intersection is between $y = \frac{1}{D^p}$ and $y = D^d$. The unique solution is clearly $x = 1$ and $y = 1$. Note that we can adjust the currency unit and the time unit so that the default equilibrium of 1 unit and an interest rate of 100% per period holds; hence, the omission of adjustable constants in the supply and demand curve formulas does not sacrifice generality.



Post-discouragement, it becomes:
$\frac{1-h}{D^p} = D^d$
$D = (1-h)^{\frac{1}{d+p}}$

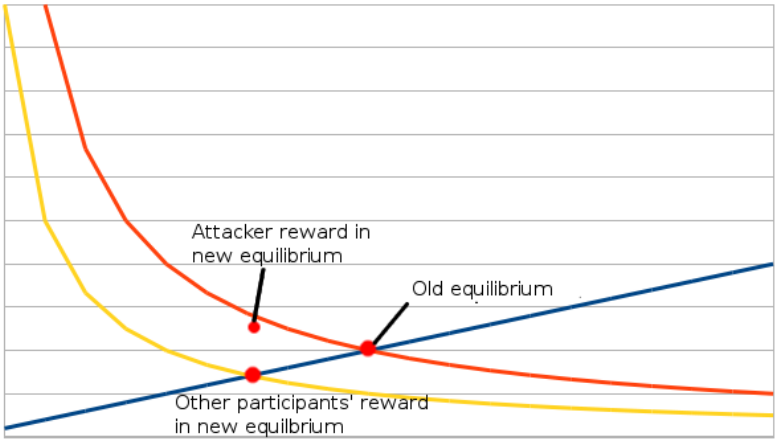


Let us now look at the attacker's interest rate, $\frac{1-\frac{h}{r}}{D^p}$. First, let us take the easy case: $r \leq 1$. In this case:

$\frac{1-\frac{h}{r}}{D^p} \leq \frac{1-h}{(1-h)^{\frac{p}{d+p}}} = (1-h)^{\frac{d}{d+p}} < 1.$

Hence, if $r \leq 1$, the attacker will always lose money. This may seem counterintuitive; one might ask, what if the discouragement attack pushes out so many other validators that the new equilibrium is on the very high part of the the supply curve close to zero? The important thing to keep in mind, however, is that if $r = 1$ (i.e. the attacker gets the same interest rate as the victims), then the attacker's revenue will necessarily be at some point along *the original, unchanged, upward sloping demand curve*. Because the demand curve is upward sloping, and the number of validators decreased, the interest rate paid to the attacker must have also decreased. If $r < 1$, then the attacker loses *even more* than the victims, at least if expressed as an interest rate, and so the attacker's interest rate will end up *below* the lower point along the demand curve experienced by victims. Hence, if $r \leq 1$, discouragement attacks are necessarily costly.

In general, it is certainly feasible to design a consensus mechanism where we can ensure $r \leq 1$ as long as the attacker controls less than 50% of validators, so this is already a very useful result. Now, let us examine the case where $r > 1$. For very high values of r, it is easy to see how the attacker can theoretically make a net gain from a discouragement attack:



However, with the right bounds we can still prevent such an attack from being profitable. Consider the case where $p = 1$, and where the attacker must maintain a 50% share of active validators to exert $r > 1$ griefing (note that at the 50% boundary, the *proportional loss ratio* r and the *griefing factor* are the same value). The next question is, does the attacker remove some of their own validators to keep their share at 50%, or do all of the validators

controlled by the attacker stay?

In the first case, as long as $p \leq 1$, no matter how high r is, the attacker's revenue must still decrease, or in the worst case where $r = \infty$, the attacker's revenue will be unchanged. In the second case, we note that the total deposit size will decline more slowly - specifically, $D = \frac{1}{2} + \frac{1}{2} * (1-h)^{\frac{1}{d+p}}$. Suppose $r \leq 2$, and $p \leq 1$. Then:

$$\begin{aligned}
&\frac{1-\frac{h}{r}}{D^p} \\
&\leq \frac{1-\frac{h}{2}}{(\frac{1}{2}+\frac{1}{2}*(1-h)^{\frac{1}{d+p}})^p} \\
&\leq \frac{1-\frac{h}{2}}{\frac{1}{2}+\frac{1}{2}*(1-h)^{\frac{p}{d+p}}} \\
&= \frac{\frac{1}{2}+\frac{1}{2}*(1-h)}{\frac{1}{2}+\frac{1}{2}*(1-h)^{\frac{p}{d+p}}} \\
&\leq \frac{\frac{1}{2}+\frac{1}{2}*(1-h)}{\frac{1}{2}+\frac{1}{2}*(1-h)} \\
&= 1
\end{aligned}$$

Hence both strategies are unprofitable. For values $r > 2$, the proof would need to be more conditional on specific values of p. We can make the claim that, if the griefing factor is bounded by GF, i.e. $r \leq GF * \frac{\alpha}{1-\alpha}$, then a discouragement attack cannot be profitable if and only if $p \leq \frac{1}{GF}$.

We can check this at the boundary $h = 1$ as follows. We want to show that $\frac{1-hp*\frac{1-\alpha}{\alpha}}{(\alpha+(1-\alpha)(1-h)^{\frac{d}{d+p}})^p} \leq 1$, so we show that the numerator is less than or equal to the denominator. At $h = 1$, the numerator simplifies to $1 - \frac{p}{\alpha} + p$ and the denominator to α^p. At $\alpha = 1$, the two are equal. To show that the numerator is strictly less for $\alpha < 1$, we can take the derivative of both with respect to α; the numerator becomes $\frac{p}{\alpha^2}$ and the denominator becomes $p * \alpha^{p-1}$, and since $\alpha < 1$ the derivative of the numerator is clearly greater, so for $\alpha < 1$ the original fraction will be less than one. Checking for $0 < h < 1$ is much harder, but analytically it can be verified that it holds.

Hence, if the griefing factor is bounded by 2, we want $p \leq \frac{1}{2}$, and similarly for other griefing factors.

3 Discouragement Attacks for Breaking Consensus

Here we evaluate the feasibility of attackers with a two-step plan. First, run a discouragement attack to push other validators out. Second, attack the network against a now much smaller validator set. The second attack could either be a finality reversion attack, or it could be censorship. In the given model, this is clearly doable: an attacker can grief with $h > 1$ to push all other validators out, then remove most of their own validators, then use the remainder to perform the attack. This can be overcome with an honest minority assumption, where some validators are willing to stay despite the lack of economic incentive, and it can also be overcome with outside donations to "honest" validators. A third way that it can be overcome is if, when such an attack starts taking place, a large number of outside players temporarily join the validator set, diluting the attacker to below 50% and thereby making their attack ineffective.

This kind of attack is difficult to economically model because under certain assumptions the cost is zero: if an attacker can credibly announce that they will grief with $h > 1$, then all other validators will leave, and the attacker will then be free to join with one single validator and perform a censorship attack at infinitesimal cost. This result is true in *any* game where the net profit of a validator can be made to drop below zero through no fault of their own, which is itself true of any consensus algorithm where a censorship attack has nonzero cost, because of the fundamental fault inattributability of censorship versus a minority going offline.

What we *can* do is model the game in various ways that add realistic "friction" to non-attacking validators' economic reasoning, and see how the parameters of the game can be optimized so as to maximize the cost of attack given these frictions. To more clearly illustrate the difference between losses on the order of security deposits and losses on the order of rewards, we now assume that all rewards and penalties are multiplied by some base interest rate y_0; that is, the victims earn $y_0 * \frac{1-h}{D^p}$ and the attacker earns $y_0 * \frac{1-\frac{h}{r}}{D^p}$.

One possibility is to model it as a three-phase game, where in phase 1 the attacker griefs with some h, all validators get their due rewards and penalties, then in phase 2 both the attacker and other validators make choices about how to allocate their resources and finally in phase 3 the attacker decides whether or not to attack.

Let us first consider finality reversion attacks. In a finality reversion attack, if the deposit size is D, the cost of an attack is $\frac{D}{3}$. An attacker's strategy is easy: grief with $h = 1$ in phase 1, drive all other validators away as their revenue drops to zero, and then attack in phase 2. The attacker's cost here, assuming the attacker had 50% of the validator set in phase 1, is $\frac{1}{2} * y_0 * (1 - \frac{1}{r})$.

Now, let us modify the game slightly: suppose that of the $\frac{D}{3}$ penalized, half goes to all other validators. The attacker griefs with some h in phase 1, and as a result in phase 2 the total deposit size drops from 1 to D_2, with base interest rate $y_2 = \frac{y_0}{D_2^p}$. The attacker then attacks with probability P_{attack}.

The attacker's cost is:

$\frac{1}{2} * y_0 * h + P_{attack} * \frac{1}{3} * D_2$

The first term in the sum is the cost in phase 1, and the second term is the expected cost in phase 2.

Supply-demand equilibrium tells us that in phase 2 we have:

$y_2 * (1 - h) + \frac{1}{4} * P_{attack} = y_0 * D_2^d$

The $\frac{1}{4}$ fraction comes from the fact that during an attack, non-attacker's deposits would increase by 25%, and because the original intersection was $(1, y_0)$ the demand curve must also be multiplied by y_0. Let us assume $d = p = 1$. We can simplify:

$\frac{y_0}{D_2} * (1 - h) + \frac{1}{4} * P_{attack} = y_0 * D_2$

Or:

$(h - 1) - \frac{P_{attack}}{y_0 * 4} * D_2 + D_2^2 = 0$

This gives us D_2 out of P_{attack} and h through a quadratic equation, which we can then plug into the attacker's cost. This gives the cost as a function of h and P_{attack}. The quadratic equation is:

$$D_2 = \frac{\frac{P_{attack}}{4 * y_0} + (\frac{P_{attack}^2}{16 * y_0^2} - 4 * (h-1))^{\frac{1}{2}}}{2}$$

The discriminant equals zero at when $\frac{P_{attack}^2}{16 * y_0^2} = 4 * (h - 1)$, or $h = 1 + (\frac{P_{attack}}{8 * y_0})^2$; if h is higher than this value then there is no intersection between the new de-facto supply curve and the demand curve, meaning that non-attacking validators will lose money regardless of what happens, and so $D_2 = 0$.

Because the benefits to the attacker of removing validators from the validator set are so high, we find that the optimal h for any given P_{attack} is generally precisely the one which sets $D_2 = 0$, ie. $h = 1 + (\frac{P_{attack}}{8 * y_0})^2 + \epsilon$.

One possible mitigation to this kind of attack is to simply make it more

difficult to grief with h much higher than 1 in the specific case where D is low. That is, suppose that there exists some behavior in the network that causes some given amount of harm to the protocol, and one cannot determine whether it is caused by offline validators or censoring validators. Instead of setting punishments proportional to $\frac{y_0}{D^p}$, set them proportional to y_0, or perhaps as a compromise $\frac{y_0}{D^{\frac{p}{2}}}$, or a piecewise function. This means that if D is low, attackers will be able to cause more disruption of performance to the network at lower cost to themselves, but in return creates a scenario where it is more difficult to engage in a discouragement attack, because causing enough damage to the network for h to exceed 1 will take a longer time.

The second case that we can analyze is the case where the attacker engages in a discouragement attack, and then in the second stage engages in a censorship attack. Here, there is no counter-pressure where validators are encouraged to stay because of the possibility they will get a windfall from the attack, as in a censorship attack all validators, including the attacker and victims, must be penalized. This case is even worse than the above, as the h required to drive out other validators will be *less* than 1. However, the mitigation strategy is broadly similar. Because this kind of attack is strictly worse than a finality reversion attack, it may not be worth the complexity to implement a scheme where malicious validators' rewards are distributed to other validators, as we can expect that malicious attackers will nearly always opt for a censorship attack instead of a finality reversion attack in any case.

4 Bribing to counter-grief

Suppose that victims ($\leq 50\%$ of the current validator set) are concerned that their revenue will decrease from y_0 to 0 as part of a discouragement attack. They can choose to bribe players who are not currently validators to enlist in order to prevent this from happening. Bribing players individually is expensive, because the bribe must overcome the player's concern that they themselves will suffer from the attack. However, with an assurance contract we can create a bribe that only works if enough players show up to properly restrain the attacker. A bribe to increase the validator set by a factor of D_n would need to pay the $D_n - 1$ newly joining players the difference between the natural supply at D_n and the natural demand at D_n.

Note that existing validators do not need to receive the subsidy, as we can design the protocol so that it is easy to become a validator but takes a

long time to leave, so they will remain validators long enough to prevent the discouragement attack (in fact, we are assuming that the current validator set are the ones *paying the bribe*).

The cost of the bribe is $(D_n - 1) * y_0 * (D_n^d - \frac{1}{D_n^p})$. If $p = d = 1$, this equals $(D_n - 1) * y_0 * (D_n - \frac{1}{D_n}) = y_0 * \frac{(D_n-1)^2*(D_n+1)}{D_n}$. If the attacker is threatening to take away the victims rewards and additionally take away portion q of their deposits, then the cost of *not bribing* is $y_0 + q$. A bribe is worth it if:

$y_0 * \frac{(D_n-1)^2*(D_n+1)}{D_n} \leq y_0 + q$

$\frac{(D_n-1)^2*(D_n+1)}{D_n} \leq 1 + \frac{q}{y_0}$

This is a quartic equation, and so has no clean solution. But we can give some approximations:

$q = 0, y_0 = 0.04 \rightarrow D_n < 1.8$

$q = 0.25, y_0 = 0.04 \rightarrow D_n < 3.36$

$q = 1, y_0 = 0.04 \rightarrow D_n < 5.7$

$q = 0.25, y_0 = 0.01 \rightarrow D_n < 5.7$

$q = 1, y_0 = 0.01 \rightarrow D_n < 10.61$

If we reduce to $p = \frac{1}{2}$, then we can increase the maximum amounts of validators we can bribe to join further, though only slightly, as for high values of D_n the cost of the subsidy is dominated by the increased reserve interest rates, not the reduced in-protocol supply. For example, with $p = \frac{1}{2}$ the maximum that it makes sense for validators to bribe in the $q = 0, y_0 = 0.04$ case increases from 1.8 to ≈ 1.87.

5 Conclusion

Discouragement attacks as a cheaper way of attacking a consensus algorithm are one of the hardest classes of attacks to come up with defenses against. This is true in proof of work as well: if a 51% attack succeeds, then there is a coordination problem opposing "honest" miners trying to recover the original fork, as none have the private incentive to participate in a fork unless everyone else does. Hence, our recommendations at this point can consist only of two parts. First, there exist marginal tweaks that can be made to mechanisms to reduce the effectiveness of discouragement, increasing difficulty of leaving the validator pool and keeping p values low (particularly by not relying solely on transaction fees) being chief among them. Second, if a discouragement attack does start happening, expect an assurance contract bringing in more

validators to be an important building block in the solution.

In general, this is still an active area of research, and more research on counter-strategies is desired.