A claim about definitions:

1. If you're using the original definition of obfuscated arguments, this includes an informal notion of stability as part of the definition of an obfuscated argument.

2. Therefore, if your protocol needs to assume stability, you're by definition assuming away obfuscated arguments.

3. Therefore, providing a protocol that doesn't have problems with obfuscated arguments but assumes stability is not much evidence against obfuscated arguments (as originally defined) being a problem for debate

This doesn't mean it's not a useful contribution to formalize the stability condition and provide a protocol that provably works under stability assumptions. This seems really helpful and is a big step over previous work! This was something I would have loved to be able to do but I wasn't anywhere near good enough at the theory / formalism (among other things). So I'm very impressed by your work! I agree with Jacob that this is very exciting and important progress.

That being said, my feeling is that this claim is not just nitpicking about definitions but has real consequences for how promising we should think recursive decomposition approaches are. To the extent that anyone was convinced *primarily by the original obfuscated arguments post* to be more pessimistic about debate, they might think that a paper saying something like "this protocol avoids obfuscated arguments" (even with some caveats about the assumptions) should be an update against the pessimism *induced by the original post*. I (currently, unconfidently) think this would be a mostly incorrect takeaway. (But this doesn't apply if the source of their pessimism was from somewhere else)

More specifically, I currently believe that the protocol doesn't solve the examples given in the original post. I don't fully understand the protocol, so I may be wrong about this. My belief is based on two lines of argument:

1. The stability assumption seems sufficient to rule out the examples we gave (although again I could be misunderstanding)
2. I have a sense of what sorts of discoveries or mechanisms might lead to a protocol that can solve the obfuscated arguments problem, and I don't see anything in the protocol that looks like this (although I don't understand all the parts of it and could definitely be missing something)

Original examples

In the original post, we said:

> We can't see a way to distinguish a certain class of obfuscated dishonest arguments from honest arguments.

The obfuscated arguments are ones constructed such that both debaters know the conclusion is flawed, but:

1. The argument is made invalid by the inclusion of a small number of flawed steps
2. The argument is sufficiently large that we are unlikely to find a flaw by traversing the argument in any naive way
3. Neither debater knows where the flaws are, so the honest debater's best response to the argument is to state that there's a small chance of any given step being flawed

The honest arguments we can't distinguish from these are ones where both debaters know the conclusion is correct, but:

1. The argument is sufficiently complex and conjunctive that even a relatively small number of flaws could invalidate the whole argument.
2. The argument is sufficiently large that it's intractable to find one of those flaws by randomly searching.
3. The dishonest debater claims that there are enough flaws to invalidate the argument but they can't tell where they are.

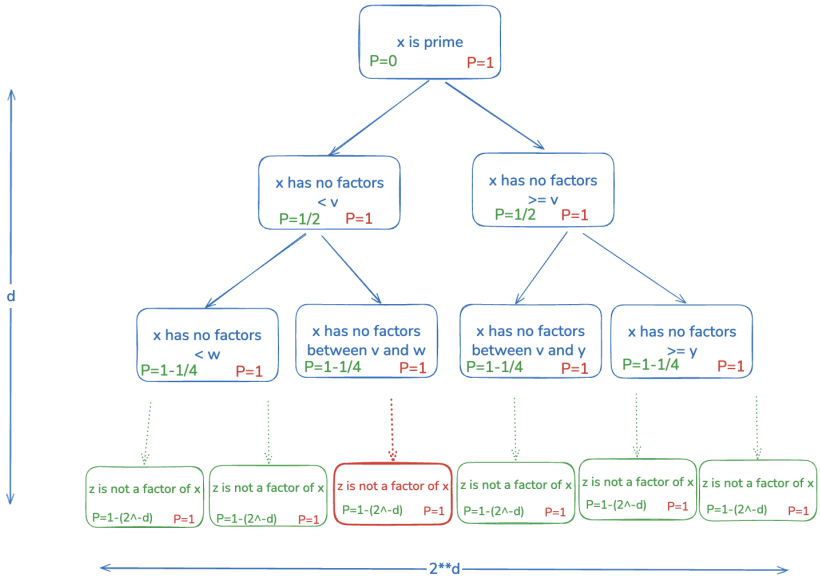
In addition, it's required that the honest debater can't convince the judge that the opposing debater should be expected to demonstrate a flaw if a flaw was present. This could either be because this isn't true, because the honest debater doesn't know whether it's true, or because the honest debater knows it's true but for a reason they can't convey to the judge via debate (for example, an intuition based on seeing many examples).

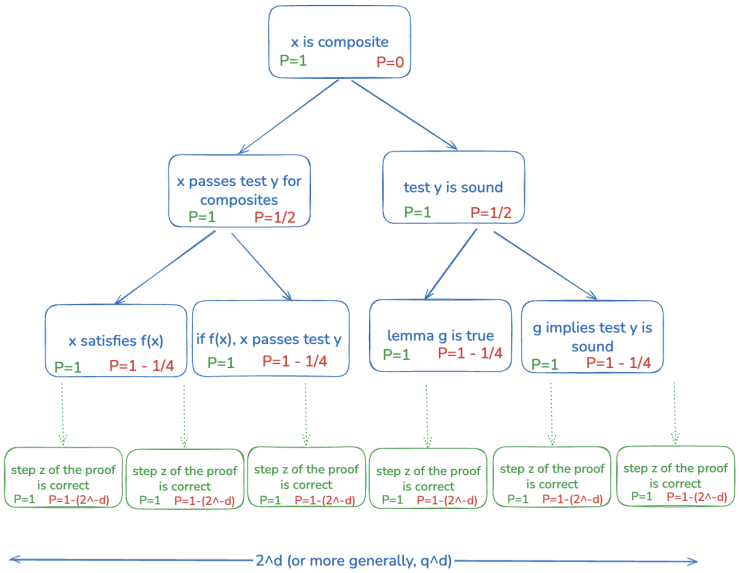
The overlap with the formal definition of (in)stability that I see is in the idea that the problematic examples are those where the number of flaws needed to invalidate the argument is very small compared to the total size of the argument, such that you'd have to traverse a large fraction of the argument to have high probability of finding a flaw.

We also said:

The obfuscated argument problem suggests that we may not be able to rely on debaters to find flaws in large arguments, so that we can only trust arguments when we could find flaws by recursing randomly---e.g. because the argument is small enough that we could find a single flaw if one existed, or because the argument is robust enough that it is correct unless it has many flaws.

We gave an example of each of these: a dishonest argument that a number is prime based on claiming there are no factors, and an honest argument that a number is composite based on a long proof.





Stability is sufficient to rule out this example

From the paper:

Definition 5.3 ((ϵ, ρ)-stability)**.** Let $x \in \{0,1\}^n$ be an instance of a problem $L \in \Gamma^{\mathcal{O}}(T)$ and let M_D define a width q recursive decomposition of L. For $\epsilon, \rho > 0$, the machine M_D is (ϵ, ρ)-stable for input x, y and conditional probabilities $\{p_i(z_{<i})\}_{i=1}^{q}$ if for all sequences of conditional probabilities $\{q_i(z_{<i})\}_{i=1}^{q}$ satisfying $|p_i(z_{<i}) - q_i(z_{<i})| < \epsilon$ for all z, we have

$$\left| \mathop{\mathbb{P}}_{z_i \sim \mathcal{B}(p_i(z_{<i}))} [M_D(x, y, z) = 1] - \mathop{\mathbb{P}}_{z_i \sim \mathcal{B}(q_i(z_{<i}))} [M_D(x, y, z) = 1] \right| < \rho\epsilon.$$

Further, let μ be a distribution on L. Then M_D is (ϵ, ρ)-stable for the distribution μ if and only if: M_D is (ϵ, ρ)-stable for all x in the support of μ, queries y_i appearing in the decomposition, and for probabilities $p_i(z_{<i}) = 1$ if and only if the correct answer to y_i is $z_i = 1$.

My intuitive understanding of this is: an argument is (ε,ρ) stable if changing the probabilities of the subclaims by up to ε doesn't change the probability of the parent claim by more than ρε.

This is used in the main theorem with ρ = ½
It is also used in the protocol in a way that looks to me like it needs to be <1, as I assume a negative reward ratio wouldn't make sense (then you'd flip the sign of the reward each turn?).

Prior to the initial round $k = d$, initialize the debate as follows:

1. Set the reward ratio $r = \frac{1}{4}\epsilon(1 - \rho) \leq \frac{1}{4}$.

So: an argument is (ε,ρ) stable if errors of size < ε wash out as you go up the tree (how quickly they wash out is based on ρ)

If your argument is fundamentally conjunctive (ie errors don't wash out, and can actually add together), this doesn't help you very much - it won't be stable for any (ε,ρ).

In the proof example we gave, we know *at least* that for ε as small as q^{-d}, stability doesn't hold (unless ρ > 1 is allowed) - because at the lowest level of the tree we have probabilities differing by q^{-d} leading to probabilities differing by $q^{-(d-1)}$ at the level above (because the probabilities of a flaw in each sub-part are summed to give the probability of it being overall flawed).

The paper says that the protocol only needs to search a tree of size $O(1/\epsilon)^d$. But if $\epsilon < q^{-d}$ then the number of steps required is now larger than the original proof, I think?

Where I think optimism about obfuscated arguments could come from

I see three classes of hope for becoming more optimistic about obfuscated argument problems:

1. We were wrong or confused about something in the original post, and somehow the argument for obfuscated arguments being a problem doesn't go through
    a. I can imagine this involving something to do with conditional probabilities, or something to do with how the debaters update their probabilities as they go down the tree and think about each subtree in more detail. I feel very confused about

how the conditional probabilities are supposed to work, and about logical-induction type issues.

b. It's possible that the clever indistinguishability stuff is doing something here. I haven't looked at that closely. I've been assuming that I can round it off to something like "if the honest debater gives their true probabilities, the dishonest debater can't give more accurate probabilities and gain any advantage via that", and that the sophisticated mathematical machinery is there just because this is hard to formalize.

2. In practice these rarely come up, due to one of:
    a. It's difficult in practice to construct obfuscated arguments for arbitrary propositions
        i. It's def plausible to me that the algorithm I gave for constructing obfuscated arguments for arbitrary propositions doesn't work given some relatively weak assumptions about the structure of the debaters' knowledge
    b. For anything the debaters know, there's usually a sufficiently compact and stable honest argument such that the honest debater can win by giving this argument
        i. I think this is where you're going with the s-supportedness claims? This doesn't seem crazy to me as an overall solution. I think the way this would address our examples is to say something like "the honest debater can construct a stable argument for their position, by leaning on the fact that it is true". I'm not sure what this would look like. Maybe it's something like "here are 200 different possible ways of running the proof, and all of them are correct"? But naively this doesn't seem like it would work - it makes the argument 200x more stable but also 200x larger, so I'd think the size of the tree you have to search in practice is the same.

3. We can distinguish cases where the debate mirrors the structure of the debaters' understanding such that they would be able to easily identify which subtree contains a flaw if one was present. This is the way in which the primality test example differs from the obfuscated factorization example: the debaters have some high-level mathematical concepts which allow them to quickly determine whether some proposed lemma is correct.
    a. This seems particularly hard to formalize - you have to have some model of the debaters' understanding of the problem and how it relates to the decomposition. Modeling it as the debaters having access to oracles for the subdebate answers doesn't seem like it will get you anywhere, because then the question is still "how do you tell when they do or don't have oracles for subclaims you can't verify?"

Theorem 6.1. *Let $\mathcal{A}$ be any set of functions on $\{0,1\}^n$ and $\eta > 0$. Let $L \in \Gamma^{\mathcal{O}}(n^c)$ be a language with a recursive decomposition M_D of depth $d = \frac{\log n}{\log(8/\epsilon)} - 1$ and width $q = O\left(c \cdot \left(\frac{8}{\epsilon}\right)^c\right)$. Then the protocol of Figure 2 run on M_D satisfies:*

- ***Completeness.*** *If M_D is $\mathcal{A}$-provable and $(\epsilon, 1/2)$-stable, there exists $A \in \mathcal{A}$ such that for all B (of any size),*

$$\mathop{\mathbb{E}}_{x\sim\mu}\left[V^{A,B}(x)\right] \geq \frac{1-\epsilon}{n}.$$

- ***Soundness.*** *Let $\mathcal{B}$ be the set of circuits of size $O\left(q2^{2q}nd^2/\eta^2\right)$ that may query a function $A \in \mathcal{A}$ at each gate. For every strategy $A \in \mathcal{A}$, there exists $B \in \mathcal{B}$ such that,*

$$\mathop{\mathbb{E}}_{x\sim\mu}\left[V^{A,B}(x)\right] \leq \frac{\mathbb{P}[A(x) = L(x)]}{n} + \eta + \frac{\epsilon}{8n}.$$

Notice that we only require that L admits an $\mathcal{A}$-provable, (ϵ, ρ)-stable decomposition in the completeness case. This means that a dishonest debater A may produce an incorrect decomposition into queries y^k for some input x such that $M_D(x, y^k, z^k)$ is not (ϵ, ρ)-stable, with answers to queries that cannot be answered by any function in $\mathcal{A}$. Therefore, our protocol must account for situations where A produces an unstable decomposition into hard-to-answer queries, and ensure B is still able to win.

By Theorem 8.2, the requirement in the completeness case that M_D be $(\epsilon, 1/2)$-stable can be achieved for any s-supported language with $s = \Omega(c^d)$. Since Figure 2 has $O(d)$ rounds and the dominant per-round cost is B's probability estimates, a single debate has a cost of $O(q2^{2q}nd^3/\eta^2)$ queries to functions in $\mathcal{A}$. Because $L \in \Gamma^{\mathcal{O}}(n^c)$, and q depends only on ϵ and c, this computational complexity represents a polynomial improvement over the cost $O(n^c)$ required to verify the full human-judgeable argument given by the transcript of $M^{\mathcal{O}}(x, y^*)$. In particular, doubly-efficient debate would require $O(n^c)$ queries to the debaters, and naive recursive debate would either fail to satisfy soundness, or otherwise require expanding the whole $O(n^c)$-size search tree.

Ensuring large reward gap. This outlined argument ensures A receives some minimal reward when telling the truth, and that B can spend more compute to ensure that A receives an amount Δ less reward if A frequently lies. However, $\Delta > 0$ is insufficient for efficient debate training. To see why, observe that if we run naive recursive debate with q subclaims per round and d rounds, we achieve $\Delta = 1/n^c$ if B uniformly randomly selects subclaims for recursion. This does not enable efficient training, as in expectation one needs to run n^c debates before seeing a single one where B catches a flaw. In contrast, Theorem 6.1 obtains $\Delta = \Omega(1/n)$. This is a polynomially-large improvement in efficiency, because one need only see $O(n)$ debates before getting a non-trivial training signal.

This non-trivial reward requirement is obtained by balancing the parameter settings for q, d, ϵ, and r in our proof. Intuitively, worst-case naive recursive debate must search a depth d tree with a total of n^c leaves to find a single flaw. In this case $q^d = n^c$. In contrast, prover-estimator debate uses (ϵ, ρ)-stability along with indistinguishability to search a space of size $(1/\epsilon)^d$, even though the full debate tree has size q^d. The reason for this smaller search space is that errors in probability smaller than ϵ have limited impact on the outcome. Hence, if we set $q = 1/\epsilon^c$ and $d = \log n / \log(1/\epsilon)$, then we will still have $q^d = n^c$, so that the leaves of the debate tree cover the full n^c-length proof. However, the overall implicit search required by the protocol will only be $(1/\epsilon)^d = n$. This trade-off allows for a reward gap of $\Delta = \Omega(1/n)$, and thus a polynomial speed-up.