An Introduction to Blockchain Mechanism Math and Terminology for Deeply Casual People who want to Follow Along and Sound Smart when Discussing White Papers

Alex Watts et al

December 12, 2024

Abstract

Mechanism Designers working in DeFi really like writing white papers. They like it a lot. *Perhaps too much.* They also like writing these papers in an ancient, hieroglyphics-based language that is difficult for a normal, socially-adjusted person to decipher. This is unfortunate because most of the math and game theory is actually quite simple once you understand their alien terminology. In this paper, I'll give you the tools you'll need to sound smart when discussing these white papers with your peers, which is a very important skill set because in all likelihood your peers are also trying really hard to sound smart.

Let's start with this section - the "abstract." This section is really just a summary of the white paper. In this section, the Mechanism Designers will summarize a problem, then they'll summarize their solution, and then - if they're good - they'll also summarize the shortcomings of the paper. We may never know *why* the Mechanism Designers like to call this section an "abstract" rather than a "summary" - in fact, we'll probably never know *why* they do most of the things that they do. But by reading this paper, hopefully you'll be better able to understand *what* they're doing - and, if you're into this kinda thing, *who* they're doing it to.

Intuitively, this paper doesn't have any shortcomings. But if I had to pick only one, it'd be that a lot of the information in this paper is deliberately wrong. *Very* wrong. I'm way off on a lot of these explanations. But I'm not being byzantine -which means "dishonest" or "adversarial", by the way, and two thousand years ago it probably would've been flagged by HR departments as racially offensive against the citizens of the Byzantine empire. So yeah, I'm not being byzantine - I'm just oversimplifying things to make the concepts easier to understand in expectation.

1 Sets

A set is a a groups of things.

By the way, if your first thought after reading that last sentence was "*that definition left out a lot!*" then be warned: *it's all downhill from here.*

- Sets are represented with an uppercase letter (such as X or Y).
- When sets are defined, they're often surrounded with brackets. For example, a set of three numbers could be written as $X = \{1, 2, 3\}$.
- Members of the set are represented with a lowercase letter (such as x or y).

There are a handful of funny-looking symbols that these Mechanism Designers like to use in their white papers that you should probably learn:

- $\in$ means "in" and is used to show membership in a set
  - When you see $x \in X$, that means that x is a member of the set X.
  - **Example:** if $X = \{2, 4, 6, 8, 10, 12, 14, 16, 18, 20\}$, then $x \in X$ means that x is one of the 10 values in X.

- $\cup$ means "union" and is meant to take two sets and combine all of their members together. This combination process overwrites their redundancies so that in the new merged set there's only one copy of each duplicate.
  - When you see $X \cup Y$, that means you're combining sets X and Y.
  - **Example:** if $X = \{1, 2, 3, 4\}$, and $Y = \{3, 4, 5\}$ then $X \cup Y = \{1, 2, 3, 4, 5\}$.
- $\cap$ means "intersection" and is meant to take two sets and make a new set only out of their redundancies so that there's nothing in the new set that didn't exist in *both* of the parent sets.
  - When you see $X \cap Y$, that means you're taking the intersection of sets X and Y.
  - **Example:** if $X = \{1, 2, 3, 4\}$, and $Y = \{3, 4, 5\}$ then $X \cap Y = \{3, 4\}$.
- $\subset$ means "sub set" - all of the elements of the first set exist in the second set.
- $\supset$ means "super set" - all of the elements of the second set exist in the first set. Note that if you see this then you're dealing with a particularly troublesome brand of Mechanism Designer and you should be on your guard for more trickery.
- : (the colon) typically means "if" and is often seen together with a funny looking upside-down A. Speaking of which...
- $\forall$ means "for all" and is meant to iterate through a set, often times in the creation of another one.
  - If you're a programmer, whenever you see $\forall$ think "for loop."
  - Often times $\forall$ and : are used to iterate through a set to create a new one.
  - **Example:** Consider this monstrous expression: $Y = \{2x,\ \forall x \in X : x > 3\}$. The english translation is something along the lines of *"Y is a set of numbers that was created by taking each of the x's in X that were greater than* 3 *and then multiplying that x by* 2*."* For posterity's sake, if $X = \{1, 2, 3, 4, 5\}$ then it follows that (aka $\Rightarrow$, as we'll discuss later) $Y = \{8, 10\}$.
  - **Warning:** When it comes to defining sets, Mechanism Designers really like to move variables around or remove them entirely. They do this not to confuse you but to confuse each other - presumably to increase their job security. Take, for example, $Y = \{2x,\ \forall x \in X : x > 3\}$. That would be the same as $Y = \{2 \cdot x \in X : x > 3\}$ or $Y = \{z \in \{2x, \forall x \in X\} : z > 6\}$. Watch out for anyone using linear algebra to define sets (look for the brackets or sets of multiple things in parentheses aka "tuples"). They're one of the least understandable species of Mechanism Designers and should be avoided by all but the most skilled larpers. In fact, if one of them is reading this right then now then they're probably day dreaming about correcting the author - presumably using made-up words like "*vector*" or "*scalar*."
- **card**(X) or $|X|$ is the "cardinality" of the set X. This is a fancy way of saying "the number of things in X." If $X = \{1, 17, 31, 5\}$ then $|X| =$ **card**$(X) = 4$. Keep in mind that $|X|$ is the size of X while $|x|$ is the absolute value of $x \in X$; be vigilant, and remember that the Mechanism Designers will only win if you let them.

There are a handful of prestigious sets that you should know:

- $\mathbb{Z}$ is the set of all integers
- $\mathbb{R}$ is the set of all real numbers
- $\mathbb{E}$ isn't a set of numbers at all - it means "the expectation of" and is used in probability. But it looks similar to these fancy sets, so be careful not to get it mixed up. That's how they get you.
- $\mathbb{C}$ is not your friend. If you see it, *run.*

The Mechanism Designers will typically use these fancy sets when defining a new set. For example, they might say $X \subset \mathbb{Z}$, which means X is a subset of $\mathbb{Z}$, which means that all of the possible x's in X must be integers. If you're wondering *"Why doesn't the Mechanism Designer just say that the set is made up only of integers?"* the answer is *because they hate you.*

2 Probability

Two things that Mechanism Designers simply can't get enough of are *intuitions* and *expectations*.

"Intuitively..." or *"The intuition is..."* means that the Mechanism Designer is about to tell you something that they think is so obvious that they won't explain it because only a moron would disagree. Unfortunately, for a Mechanism Designer i in the set of all people P, the set of morons $M = \{p, \forall p \in P : p \neq i\}$. If you want to understand a Mechanism Designer's *intuition*, your best chance is to take their culture's traditional approach of asking for an explanation after you beat them in duel with flint-lock pistols. Intuitively, you should be sure to ask quickly.

"The expectation is..." or *"... in expectation."* means that the Mechanism Designer is about to do some math, and we can expect that the math will probably involve probabilities.

- $P(y)$ is the probability of event y occuring. Example: for a coinflip, $P(\text{heads}) = 0.50$
- $P(y \cap z)$ is the probability of event y *and* event z both occuring. This is also abbreviated as $P(y, z)$.
- $P(y \cup z)$ is the probability of event y *or* event z occuring.
- $P(y|z)$ = the probability of event y occuring *if* we assume event z has already occured. For example, assume a coinflip, but this time there is a cheater using a weighted coin that can change the odds from 50-50 to 80-20. In that case, $P(\text{heads}|\text{cheater bet on heads}|\text{cheater's flip}) = 0.80$ and $P(\text{heads}|\text{cheater bet on tails}|\text{cheater's flip}) = 0.20$.
- $\mathbb{E}[X]$ = The expected (or probabalistic, if you want to sound smart while actually being wrong) value of X. For example, if there's a game with a coinflip and you get \$0 for tails and \$1 for heads then $\mathbb{E}[\text{game}] = \$0.50$. Note that if we want to treat the game's outcomes as a set, $E[X] = \{E[\text{heads}],\ E[\text{tails}]\} = \{\$0, \$1\}$, and $\mathbb{E}[X]$ is just the average of the expected values (aka "the expectation of") the possible outcomes in X.
- $\mathbb{E}[X|y]$ = The value of X in expectation if we assume that y happened. For example, if there's a game with a coinflip and you get \$0 for tails and \$1 for heads, but there's a cheater who is using a weighted coin that can change the odds from 50-50 to 80-20 (against you) then $\mathbb{E}[\text{game}|\text{cheater's flip}] = \$0.10$.
- Now imagine that the cheater gets to flip some of the time and you get to flip the coin the rest of the time:

$$\mathbb{E}[\text{game}] = (\mathbb{E}[\text{game}|\text{cheater's flip}] \times P(\text{cheater's flip})) \ + \ (\mathbb{E}[\text{game}|\text{your flip}] \times P(\text{your flip}))$$

- $\mathbb{P}(x)$ is just a fancy way of saying $P(x)$. Some Mechanism Designers insist that if $P(x)$ is the probability of x, $\mathbb{P}(x)$ is the probability of x *in expectation*, but nobody knows what that means and so we just ignore them and move on with our lives.

3 Fancy Math

$\sum$ Sum

This sigma may have been an key part of your fraternity or sorority's identity during college, but it has another, less-important use case: math.

$$\sum_{x=1}^{n} f(x) = \text{ the sum of } f(x) \text{ for all } x \text{ values from 1 to } n$$

In other words, it's a "for loop" that sums the different values of $f(x)$, with x ranging between 1 (the bottom) and n(the top).

Math Example:

$$\sum_{x=1}^{4} 2x = 2 + 4 + 6 + 8 = 20$$

Note that you can replace the range notation of $\sum$ with a set. If $X = \{1, 2, 3, 4\}$ then

$$\sum_{x \in X} 2x \;=\; 2 + 4 + 6 + 8 \;=\; 20 \;=\; 2 \sum_{X}$$

Mechanism Designers really like talking about whether x should start at 1 or 0, although nobody knows why. Leading experts in the study of Mechanism Designers have hypothesized that it's a core part of their mating ritual, but the results are still inconclusive.

$\prod$ Product

This *"pi from the uncanny valley"* is actually a product:

$$\prod_{x=1}^{n} f(x) = \text{ the product of } f(x) \text{ for all } x \text{ values from 1 to } n$$

The best way to explain it is through a comparison:

$$\sum : \prod :: \text{addition} : \text{multiplication}$$

If you don't remember the : :: : comparison format from the SATs then you are beyond saving.

$\bigcup_y^x$ or $\binom{n}{k}$

Unless x and y are both small, normal-looking numbers, you're about to have a *really* bad time. The math isn't hard, it's just a real pain in the ass to write out. The one on the left is an iterator for the union of sets and the one on the right is the binomial coefficient. If you see either one, *buckle up*, because you're about to combine a lot of sets.

$\frac{d}{dx}$ Derivative

Get excited because it's finally time for everyone's favorite subject: calculus!

$$f(x)\frac{d}{dx} = f'(x) = \text{the derivative of} f(x)$$

Derivatives measure the rate that one thing is changing (probably x) relative to the rate another thing is changing (probably y, but maybe t if the Mechanism Designer is fully domesticated). If $f(x)$ is a line, then its derivative is the slope of the line. In other words, it's the rate of change of the line. If there is a line that graphed "time" on the x-axis and a car's distance from the starting point on the y-axis, then the derivative of that line would be the car's velocity (the rate that the position is changing relative to the rate that time is changing). If the velocity of a car is on the y-axis, then the derivative of that line would be the car's rate of acceleration. This is what they teach in calc 1, but that you haven't needed to use since highschool because it only recently became cool to discuss "novel mechanisms." It looks like your teacher was right all along.

$\int$ Integral

This squigly line is an "integral." Fun fact - there are no integrals in the bible.

$$\int_y^x f(x) = \text{the integral, aka "antiderivative."}$$

If a function $f(x)$ creates a line on a graph, its integral is the area underneath it. Even your highschool teacher would've admitted that it's unlikely that you'll need to use integrals in your day-to-day job. After all, integrals are pretty useless unless you're either a math teacher or having to deal with the probabilities of expected probabilities in a Mechanism Designer's white paper. Speaking of which...

4 Back to Probabilities

$F_X(x)$ Cumulative Distribution Function

$F_X(x)$ is a cumulative distribution function, aka CDF. If you have a distribution X (which is a set of all possible values that x could be) then $F_X(x) = \mathbb{E}[P(x > X)]$ = The probability that x is greater than a randomly selected value from the set X.

Example: If $X = \{2, 4, 6, 8, 10, 12, 14, 16, 18, 20\}$ and $x = 8$, then $F_X(8) = 0.30$ because when you draw a random number from X there is only a 30% chance that you'll get one of the three that are less than 8 (2, 4, and 6).

$f_X(x)$ Probability Density Function

$f_X(x)$ is a probability density function, aka PDF. It is basically saying the probability that a randomly chosen value from X will equal x.

Example: If $X = \{2, 4, 6, 8, 10, 12, 14, 16, 18, 20\}$ then $f_X(6) = 0.10$ because there's a 10% chance that we'll draw a 6 from the set. In other words, $f_X(x) = \mathbb{E}[P(x = X)]$. Don't think about that equation too much - if the thought of $x = X$ seems contradictory to you, that's a good sign that you're still a healthy, normal person.

Example2: $X = \{2,\ 2,\ 6, 8, 10, 12, 14, 16, 18, 20\}$ (note that we replaced the 4 with another 2) then $f_X(2) = 0.20$ and $f_X(4) = 0.0$.

Note that $F_X(x)$ is very useful in analyzing auctions because if X is the set of all bids, $F_X(x)$ is the probability that our bid x is greater than a randomly selected bid, and $F_X(x)^n$ is the probability that our bid x is greater than n number of bids.

The calculus from the previous section comes into play because the probability density function $f_X(x)$ is the derivative of the cumulative distribution function $F_X(x)$, and the cumulative distribution function is the integral of the probability density function:

$$f_X(x) = F_X(x)\frac{d}{dx}$$
$$F_X(x) = \int_{-\infty}^{\infty} f_X(x)$$

We'll often have to use calculus to get back and forth between how likely someone is to bid something, and how likely a bid is to be higher than another bid.

5 Auction Hieroglyphics

People typically refer to the set of players (so-called) in an auction as P, and a player in the set of players as i. Nobody knows why i was chosen over p, but it was probably so that Mechanism Designers could go around saying *"i player"* and to each other and laughing at their clever inside joke. This has been going on for *decades*.

If the bids are referred to as b then b_i would be i's bid. If you want to compare two players, j is typically a stand-in for "the other player," whereas $-i$ is the stand-in for "all players other than i."

Symbols Over (or under) Letters

Sometimes a Mechanism Designer might want to share a new formula with you that is similar to an existing one, but that's just *slightly* different. If you see weird symbols over or under letters, it probably means the equation or variable has had something added to it to make it *extra* special. Here are some examples:

- If i is a player (bidder) in an auction, i' might be his sworn nemesis.
- If b_i is the bid of player i, b_i^* might be the *optimal* bid.
- If $g(x)$ is a function that works for everyone, $g_i(x)$ is a function that only works for special players like i.
- If t^2 isn't meant to be t squared, it may mean that it's the second t in a sequence of t's. Maybe the 2 is on the bottom, but then where would we put the i to mark the t as special? This is a perfect example of the kind of difficult question that Mechanism Designers spend most of their time on.

ϵ Epsilon

Mechanism designers use the ϵ (epsilon) symbol a non-trivial amount, which is ironic because it represents a trivial amount. Trivial, by the way, is just a cooler way of saying *"really tiny."* A Mechanism Designer might say something along the lines of "the optimal bid value was market price less epsilon": $b_i^* = v_i - \epsilon$.

$\cdot$ The Dot Thingy

While this may mean multiplication, if you see it by itself inside of a function then it probably means the Mechanism Designer is being lazy and didn't want to copy and paste their math. You'll typically see this only after you've already seen the full version. For example, if you're unlucky enough to see something like $y = z + g(x^2 + \mathbb{E}[Z] - \epsilon)$, then later on you might see $a = 2z + g(\cdot)$, where the $\cdot$ is a stand-in for $x^2 + \mathbb{E}[Z] - \epsilon$.

$\Rightarrow$ Therefore

If the Mechanism Designer wants to prove *why* something is the way that it is, they might use this arrow thingy. For example, if a Mechanism Designer wants to show that size of his mechanism proves that he's really smart, it might look like:

$$\mathbf{card}(mechanism_i) > \mathbf{card}(mechanism_j) \forall j \in P : j \neq i \Rightarrow F_{IQ}(iq_i) = 1 - \epsilon$$

A good exercise is to assess whether or not you actually understood that equation. If you did, it means you've been paying attention to the paper! Unfortunately, it also means you're less cool in expectation.

$\phi, \theta, \gamma, \delta, \sigma, \psi, \tau,$ *etc...* Greek Letters

Mechanism Designers love defining variables. Unlike mathematicians, Mechanism Designers really like for their variables to be *exotic*, and so they'll often use lower-case greek letters. It's a best practice to always have a Greek alphabet available when reading a mechanism's white paper so that you can quickly check to see if the designer is referring to a variable - which is pretty normal - or using some sort of ritualistic-sacrifice-based summoning math - which is a red flag. Intuitively.

6 Smart-Sounding Auction Words

ex ante* and *ex post

When auction players have to figure out what their bid for an item is before they know what the value of the item is, the auction is said to be *ex ante*. If the value of the item is known at bidding time, the auction is said to be *ex post*. This is one of the rare cases in which the obscure auction language is actually less wordy than what its describing. Way to go, Mechanism Designers!

First-Price and *Second-Price*

A first-price auction is one in which the highest bidder pays what they bid. A second-price auction is one in which the highest bidder pays what the second-highest bidder bid. Mechanism Designers really like second-price auctions because they pay the beneficiary more than a first-price would, but they're also easier to cheat and therefore require a more robust mechanism.

Warning: Never ask a mechanism designer why second-price auctions are better than first-price. It's like asking your wife about any interesting dreams she's had recently, or what kind of problems have been caused by that girl at work she doesn't like. If a Mechanism Designer ever brings up the subject of first-price vs second-price, just look them directly in the eyes and say "With a properly designed mechanism, second-price leads to more auction revenue in expectation, obviously!" and then change the subject. This response is a strictly-dominant strategy.

Sealed Bid means "private"

Moving along...

Utility Function

A utility function, often called a payoff function, is how valuable someone thinks something is. It's typically measured *in expectation*, but this was developed by the French so the $\mathbb{E}$ is silent.

Example: If $U(x)$ is a utility function, then $U_i(b_i, b_{-i})$ is the payoff to player i considering their bid (b_i) and their competitors' bids (b_{-i}).

Bid Shading

Bid shading occurs when bidders bid less than their true value - in other words, $v_i > b_i$. This typically happens because the bidder can make more money by bidding below their true value. Or, in the native tongue of Mechanism Designers, *"The utility function of bidders is optimized when their private valuations exceed their equilibrium bid."*

Incentive Compatibility

Something is considered incentive compatible (IC) if the participants in an auction are willing to bid their true value. Every bidder i assigns a value v_i to the item. The mechanism is incentive compatible when $v_i = b_i$. Mechanisms can be strongly incentive compatible or weakly incentive compatible, but nobody really knows what this means. It probably has something to do with how big the mechanism is.

Bayesian-Nash Equilibrium

A Bayesian-Nash Equilibrium (or, as the kids say these days, a "BNE") exists when there are multiple rounds and each bidders' optimal strategy (AKA their "Best Response" or br_i or s_i^*) stays the same for each round. In other words, they won't benefit from using any "trick plays" or "bamboozles."

Note that a mechanism with incentive compatibility is considered significantly better than a mechanism with just a bayesian-nash equilibrium. If two Mechanism Designers meet in the wild and one of them has an IC mechanism and the other has a BNE mechanism, then the wife and children of the BNE designer will instinctively join the tribe of the IC designer and the BNE designer will have to start over with building his mechanism and family. Mother nature is cruel, but efficient.

Credible Neutrality

Nobody *actually* knows what credible neutrality means. This has led to many Mechanism Designers making up their own definitions, presumably using their intuition. Unfortunately, these different definitions of *credible neutrality* are not credibly neutral, which is why the term remains undefined.

Fair Exchange Problem

The *fair exchange problem* refers to the difficulties of getting the parties of a trade to actually do what they promised they'd do, such as paying a bid or selling at an asset at a certain price. Mechanism Designers *love* making people do things, so this is one of their favorite problems to solve.

When someone refers to a *free look problem*, know that you're talking to an actual human being - a Mechanism Designer will always out themselves by calling it a *fair exchange problem*. Usage of the term *"optionality"* is a red flag but still inconclusive.

7 Conclusion

The conclusion sections of a Mechanism Designer's white paper is a result of feeding the rest of their paper into a LLM model and then asking it to generate a summary. This is why nobody ever actually reads the conclusion section and neither should you. I'd also like to thank my co-author, Et Al.