This is a writeup of discussions between myself,@adrian, and Eli Krenkzke.

The basic concept is to design a proof-of-stake system with automated market making based on a target maturity profile. Let's say that we start with a target maturity profile curve (set somehow, e.g. with governance), which specifies what fraction of the outstanding bonds the system would prefer to have at each maturity date:

(imagine that this graph is normalized)

Call this function maturity_pref

.

Finally, let's say that we have a current maturity profile for outstanding bonds in the same range as the graph above. Then, we can design a bond AMM which will allow conversion between bonds of different maturity (where unbonded stake is equivalent to 0-maturity).

New bonds are issued at constant price of 1 (as in current proof-of-stake systems). Interest rates for new bond issuance will be set based on the difference between the current and desired maturity profile curves (varying over time to target the desired maturity profile).

Prices for repurchases of old bonds (paying accrued interest but forfeiting future interest) will be set based on the difference between the current and desired maturity profile curves, with perhaps some additional penalty (reverse-D-controller) when the rate of change is high (to discourage rapid change).

Compared to current proof-of-stake systems, I think this design has several benefits:

- Much more flexible economic controls
- Simple UX, no third-party exchange of bonds necessary

It also raises some questions:

- How should these targets be set?
- If third-party exchange does emerge, how might it operate?
- How should voting weights be set based on outstanding bonds? Should future interest be incorporated (e.g. higher weights to longer-maturity bonds)?

Relevant links:

- Yield curve control