BLOCKSCIENCE

Closing The Loop: Documenting Model Selection

October 29, 2020

For Reflexer Labs

Outline

I. Overview of model trajectory to date

A. First version: understanding market response

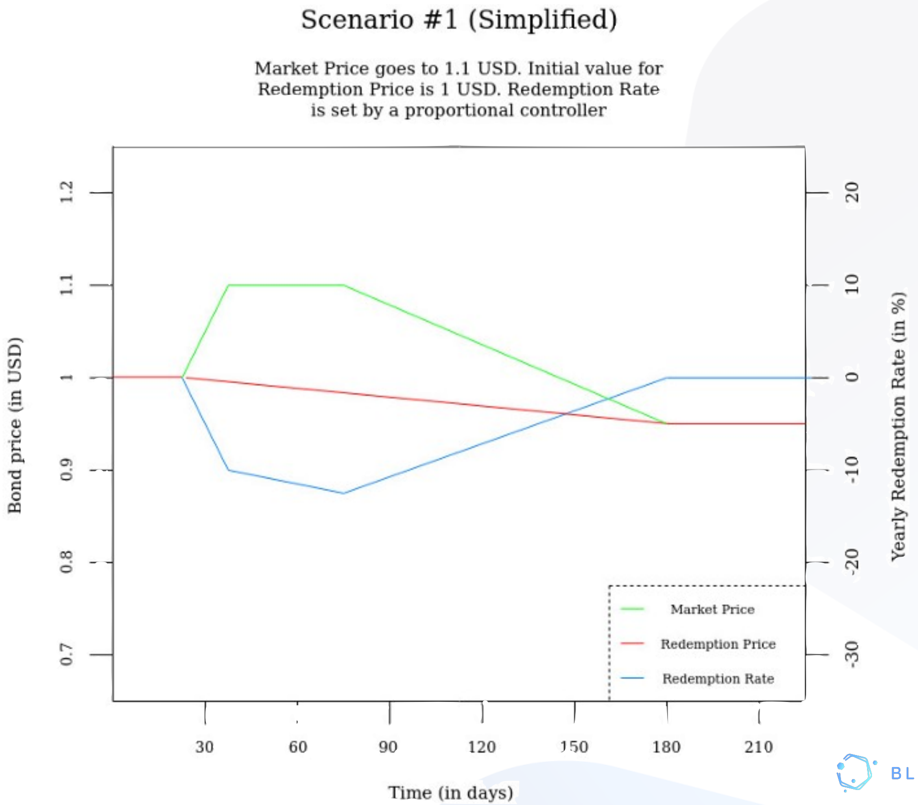
B. Second version: understanding debt mechanism

C. Third version: closing the loop with an endogenous debt mechanism and pricing theory

II. Guidance from cadCAD/Solidity Code comparisons

Model Development Overview: First Version

- Introduced a PID controller to stabilize differences between market price and a reference target, using redemption price
- Goals: e.g. recover reference target following shock, reduce volatility, prevent liquidity cascades

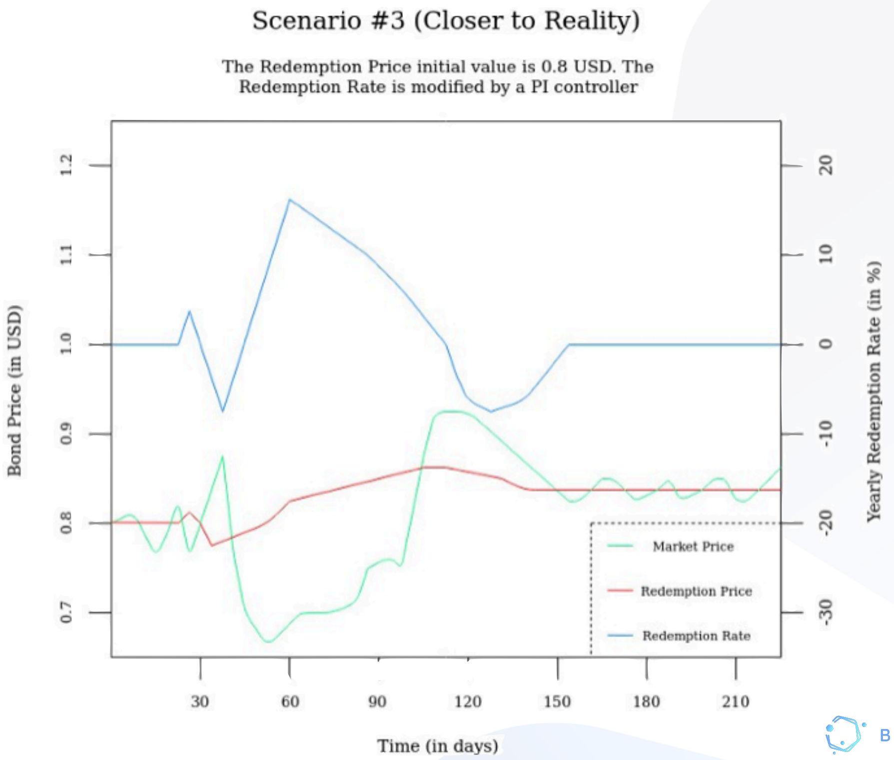


Source: [Stability Without Pegs]

BLOCKSCIENCE

Model Development Overview: First Version

- Introduced a PID controller to stabilize differences between market price and a reference target, using redemption price
- Goals: e.g. recover reference target following shock, reduce volatility, prevent liquidity cascades

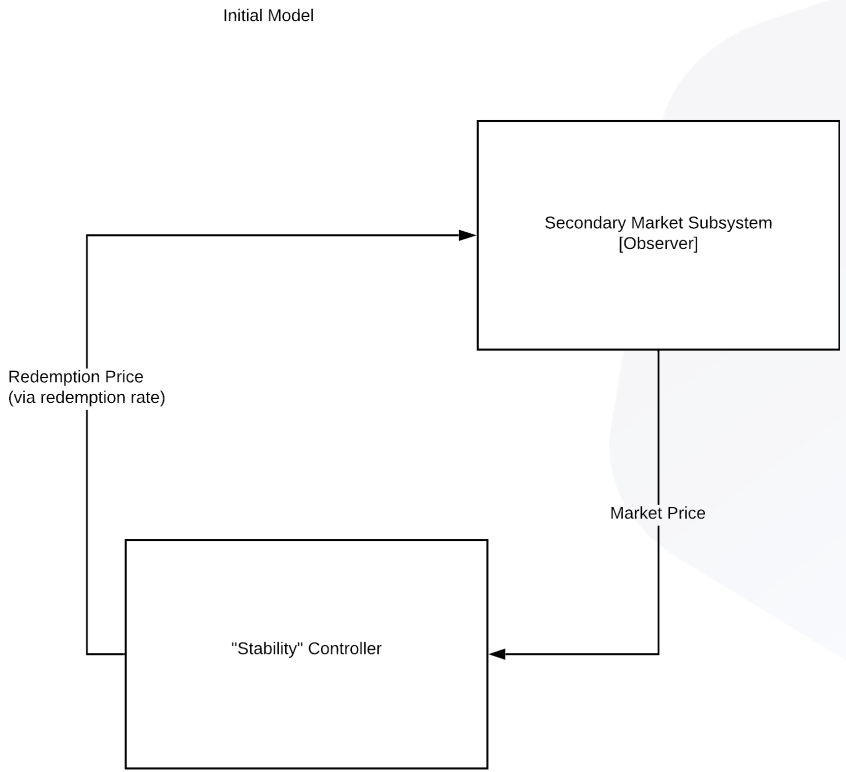


Source: [Stability Without Pegs]()

BLOCKSCIENCE

Model Development Overview: First Version

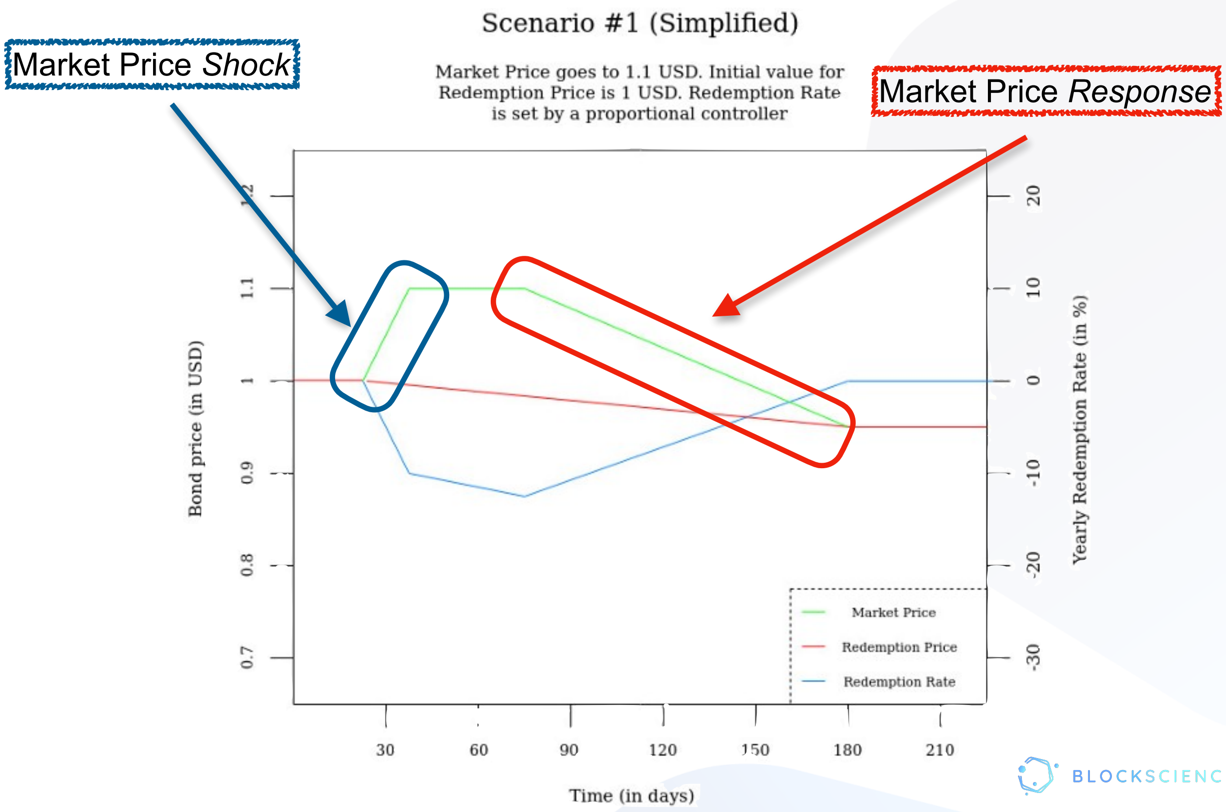
- Introduced a PID controller to stabilize differences between market price and a reference target, using redemption price
- Goals: e.g. recover reference target following shock, reduce volatility, prevent liquidity cascades



BLOCKSCIENCE

First Version: Market Response

- Market price shock is exogenous (e.g. liquidity trading)
- Market price response is assumed



Controller Specification

- P(ID) Controller: vary *control* according to *set point* to influence plant *output* using output-input *error;* output then affects system *input (sensor)*
  - $p_s(t) \equiv p_r(t)$
  - $e(t) = p_r(t) - p(t)$
  - $r(t) = K_p \cdot e(t)$
  - $p_r(t + dt) = (1 + r)^{dt} \cdot p_r(t)$
  - $p(t + dt) = F(p_r(t + dt)\,; \ldots)$
- First version: F unspecified

Notation

$p(t)$ = market price (sensor, process var)

$p_s(t)$ = set point, reference price

$p_r(t)$ = redemption price (output)

$r(t)$ = redemption rate (control)

$e(t)$ = pricing error (error)

dt = redemption price update interval

K_p = proportional parameter

Example: Linear Market Response

- Unknown F: redemption price impact on market price is assumed (not modeled) => can accidentally assume what is desired
- Example: $p(t+1) = p_0 + \alpha p_r(t+1),\ \alpha \in (0,1),\ p_0 > 0$
- Reduced form dynamics:

$$p_r(t+1) = \left(1 + K_p\left[p_r(t) - (p_0 + \alpha p_r(t))\right]\right) p_r(t)$$

- Steady-state: $\bar{p} = \bar{p}_r = \dfrac{p_0}{1-\alpha},\ \bar{e} = 0$
- $\bar{p}$ is globally stable if

$$\left|1 + K_p p_0\right| < 1$$

- Indeterminacy: *any* $0 > K_p > -\dfrac{2}{p_0}$ will do

BLOCKSCIENCE

Example: Linear Market Response

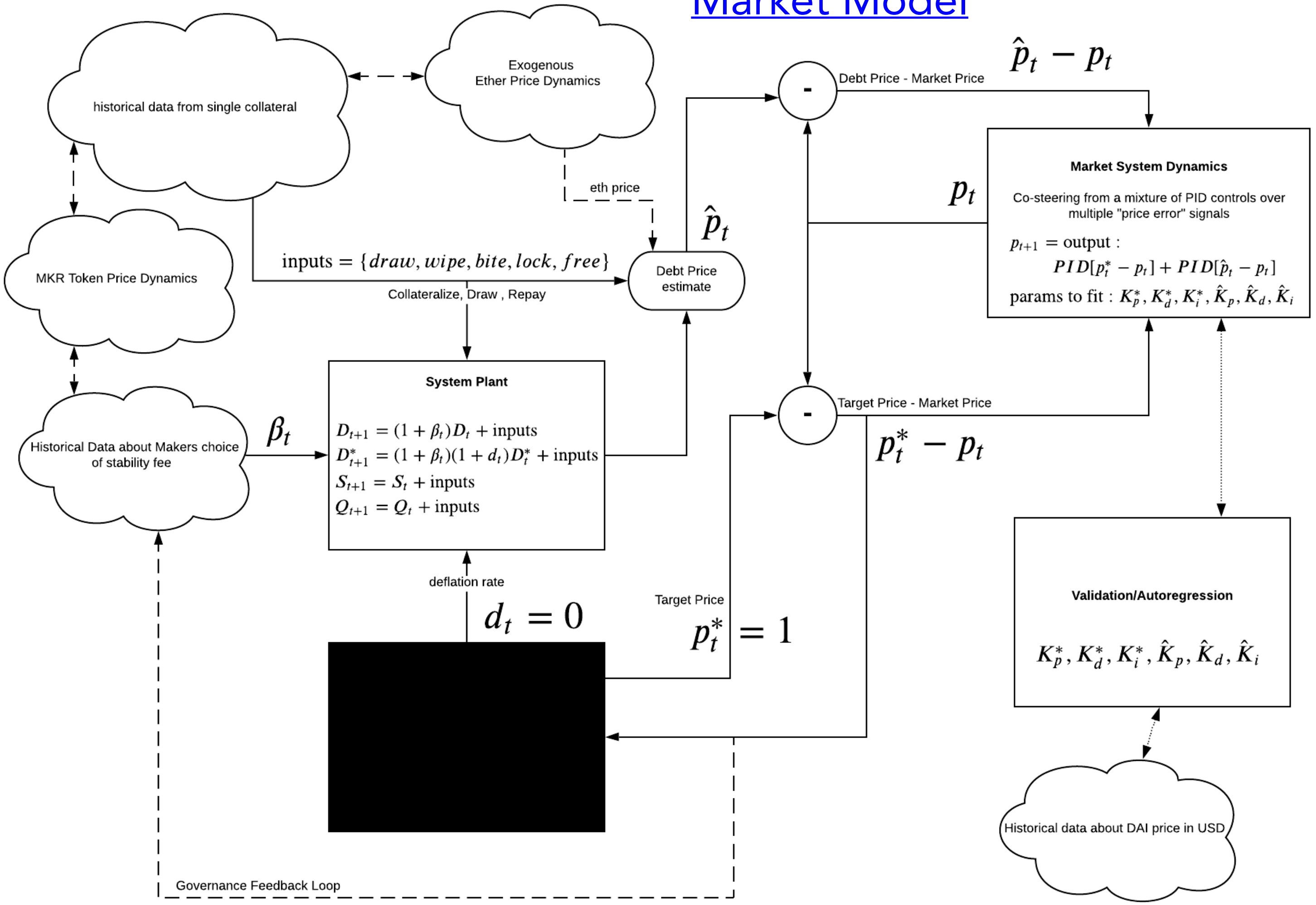
- Example: $p(t+1) = p_0 + \alpha p_r(t+1)$, $\alpha \in (0,1)$, $p_0 > 0$
- Positive K_p? Suppose $\dfrac{2}{p_0} > K_p > 0$.
- Without integrator term K_i, leads to:
  - stable value of error term (non-zero)
  - stable value of redemption price (different from market)
  - stable value of market price, but ***not*** $\bar{p}$
- New market price: p_o, which *may be a target price*
- Structure of secondary market assumption governs ***stability***: sign/value of K_p governs ***steady-state price***

BLOCKSCIENCE

First Version Updating

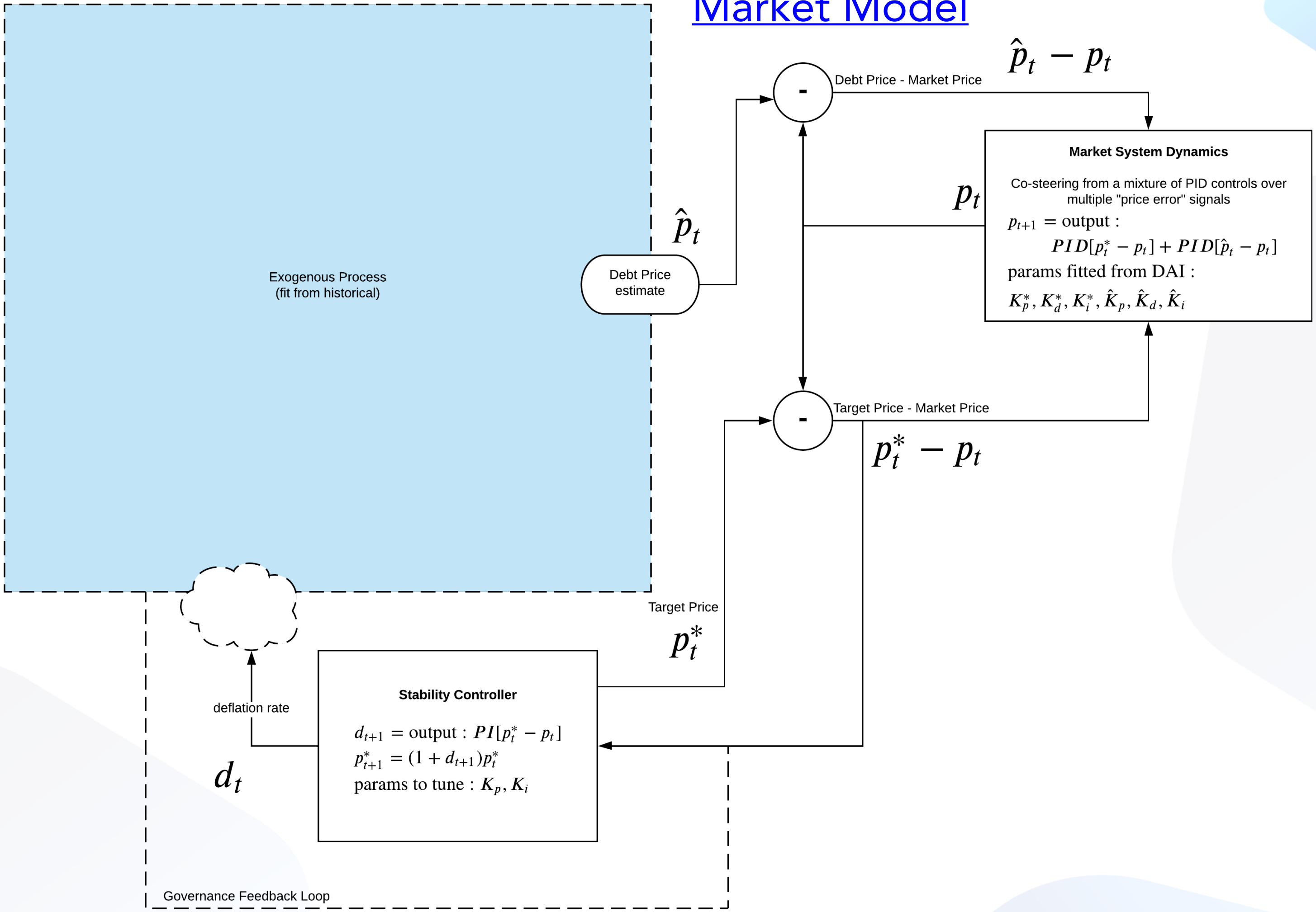
- Move away from *ad hoc* specification of F
- Understand instead impact of redemption price on the mechanism where ETH is converted into RAI via *debt issuance*
- Create a model of the collateralized debt position (CDP) mechanism, recognizing ETH price p_e is an exogenous driver
- Generate an imputed cost of borrowing: “debt price” p_d using Machine Learning on Exponential Moving Average process
- *Model* the market price p as a “reflection” of debt and redemption prices, and own prior values (autoregression)

Market Model



BLOCKSCIENCE

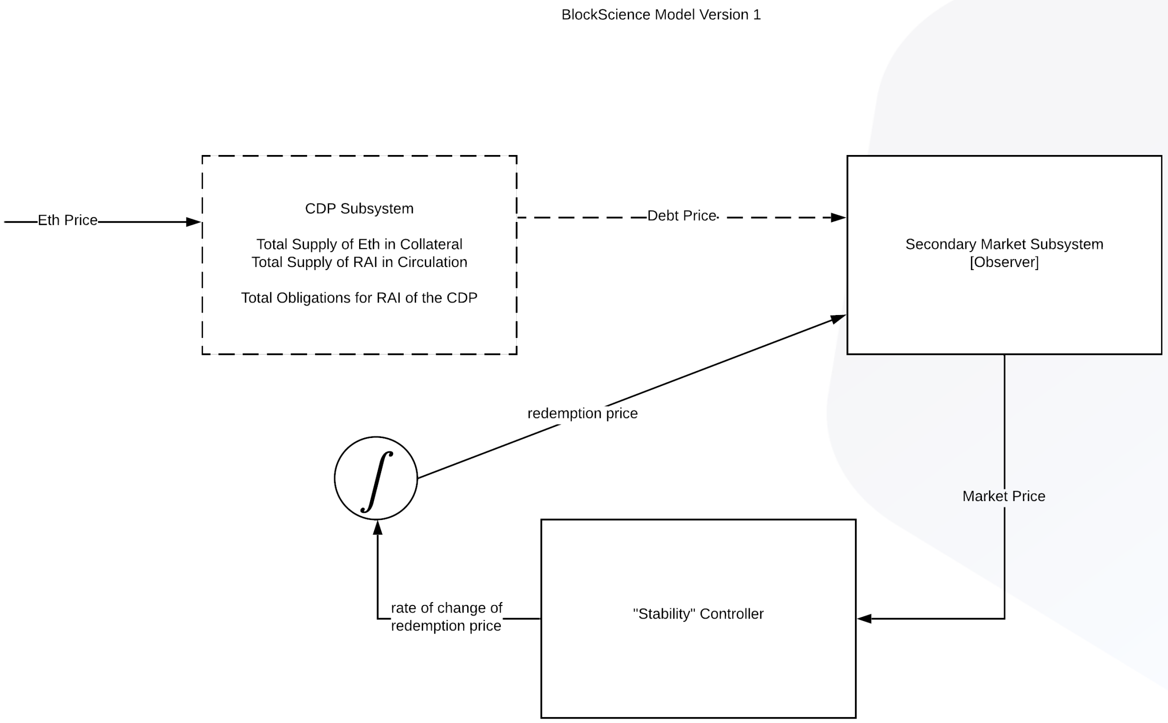
Market Model



BLOCKSCIENCE

Model Development Overview: Second Version

- BlockScience Model v. 1.0
- Experimentation: posit exogenous processes for debt price and market price, and **test stability controller**



BLOCKSCIENCE

Second Version Stability Testing

- Specified exogenous shock processes for market price p or debt price p_d; used for K_p parameter sweeps & testing stability of (the stability) controller
- Set point is *still* the redemption price p_r
- Stability here means of the controller: *bounded error* between redemption price p_r (set point) and p, other KPIs
- When p is an exogenous shock process:
  - *cannot* test **system stability** if p is exogenous; p cannot respond to p_r
  - *can* test **controller stability** by keeping error bounded

BLOCKSCIENCE

Second Version Stability Testing

- When p depends upon p_r, p_d and previous p (regression)
  - p_r does double duty: affects p *directly* via regression, but *indirectly* via p_d
  - Example: redemption rate $r < 0$ => decrease p_r and increase p_d => overall change in p ambiguous (Github repo Market Price Regression notebook)
  - Can lead to counterintuitive (p_r, p) relative movements when testing, because p is a regression without a formal connection to the CDP
  - Used only to test stability of **controller** (error bounds & other KPIs)

BLOCKSCIENCE

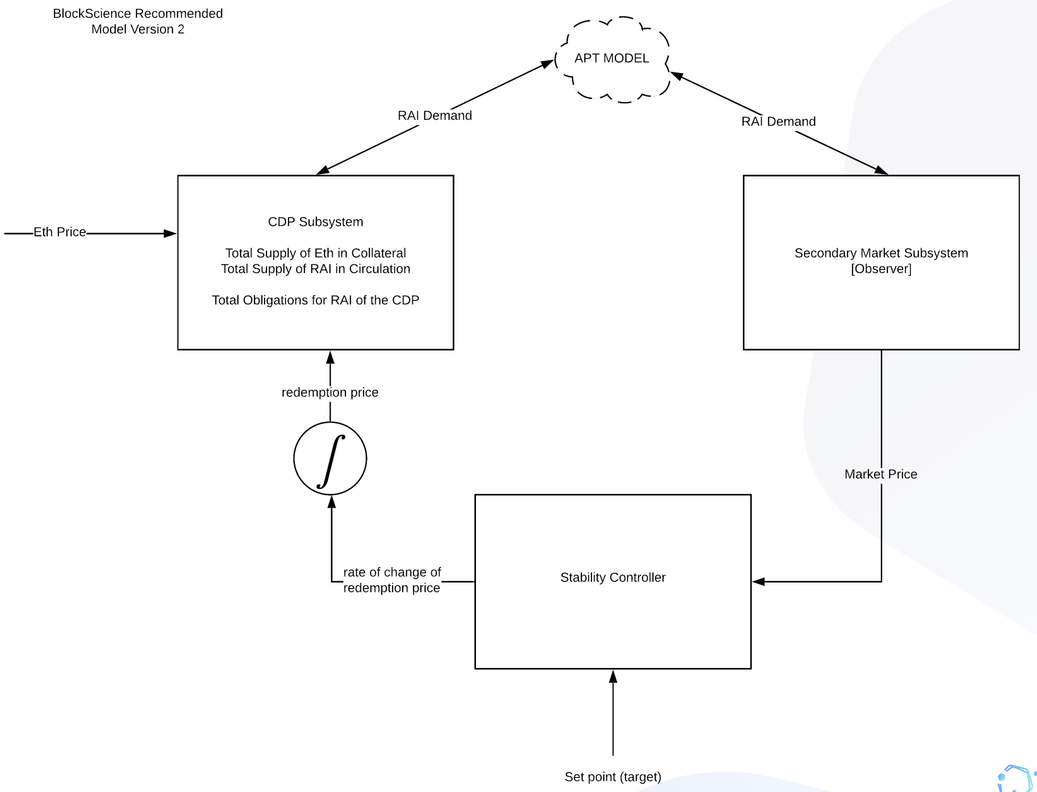
Second Version Updating

- Response of p to movements in redemption price p_r is still unmodeled
- No explicit updating of p_d in response to changes in redemption rate r or stability fee β: p_d modeled as moving average of p from which features are extracted
- Cf. Github repo Market Model, machine learning Debt Price notebook

BLOCKSCIENCE

Model Development Overview: Third Version

- BlockScience Model v. 2.0
- Closed loop system: CDP and Secondary Market are brought fully into the mechanism



BLOCKSCIENCE

Third Version Closed Loop System

- Set point p_s is independent of p_r, and is an explicit target signal
- Redemption rate r and stability fee β are brought explicitly into debt price mechanism (cf. Next Steps Notebook)
- Secondary market is explicitly connected to CDP system using Arbitrage Pricing Theory (Ross, 1976a, 1976b)
- APT allows the market price p to act as originally envisioned: a sensor measurement that is input to the controller
- p is a measurement of the unobservable **debt price** p_d

BLOCKSCIENCE

Market Price as a Measurement

- Redemption price p_r double duty:
    - acts as direct cost of borrowing: converting ETH into RAI
    - acts as unit of debt repayment: burning RAI to reduce debt
- But total cost of borrowing, p_d, includes direct cost *and* indirect cost (risk associated with ETH price p_e changes, risk associated with exogenous RAI demand changes on secondary market, etc.)
- APT: p "tracks" p_d as traders change positions to eliminate arbitrage opportunities

BLOCKSCIENCE

Third Version Stability Controller

- Stability controller [as PI(D)] acts as intended:
  - Redemption price is no longer the set point of the controller
  - Redemption rate is still the control
  - Stability fee *can* be introduced as a control
  - Output is still the redemption price
  - Input/process variable is the ***observed market price*** that tracks the unobserved debt price via the APT
  - Stability of market price is now **governed** by the stability controller => should lead to tight recommendations for K_p, K_i

BLOCKSCIENCE

cadCAD/Solidity Comparisons

- Continued 'sanity checks' to ensure responses in both systems are similar
- Continued standardized exogenous shock representations
- Third version: can "turn off" various endogenous variables and use shocks (as before) & KPIs
- Third version: can use real exogenous data from ETH price and secondary market order book (if available)

BLOCKSCIENCE