Rai: uma baixa volatilidade, garantia minimizada de confiança para o ecossistema DeFi

Stefan C. Ionescu, Ameen Soleimani

Maio de 2020

Resumo. Apresentamos um protocolo de governança minimizado e descentralizado que reage automaticamente às forças de mercado para modificar o valor-alvo de seu ativo colateralizado nativo. O protocolo permite que qualquer pessoa aproveite seus ativos criptográficos e emita um "índice reflexo", que é uma versão amortecida de sua garantia subjacente. Descrevemos como os índices podem ser úteis como garantia universal de baixa volatilidade que podem proteger seus detentores, bem como outros protocolos financeiros descentralizados, de mudanças repentinas no mercado. Apresentamos nossos planos para ajudar outras equipes a lançar seus próprios sintéticos, aproveitando nossa infraestrutura. Finalmente, oferecemos alternativas para oráculo atual e estruturas de governança que são freqüentemente encontradas em muitos protocolos DeFi.

Conteúdo

Introdução O

dinheiro é um dos mecanismos de coordenação mais poderosos que a humanidade alavanca para prosperar . O privilégio de administrar o suprimento de dinheiro tem sido historicamente mantido nas mãos da liderança soberana e da elite financeira, enquanto é imposto a um público em geral inconsciente. Onde o Bitcoin demonstrou o potencial de um protesto popular para manifestar um ativo de mercadoria de reserva de valor, o Ethereum nos oferece uma plataforma para construir instrumentos sintéticos lastreados em ativos que podem ser protegidos da volatilidade e usados como garantia, ou atrelados a um preço de referência e usado como um meio de troca para transações diárias, tudo reforçado pelos mesmos princípios de consenso descentralizado.

O acesso sem permissão ao Bitcoin para armazenar riqueza e instrumentos sintéticos adequadamente descentralizados no Ethereum estabelecerá as bases para a revolução financeira que se aproxima, fornecendo àqueles que estão à margem do sistema financeiro moderno os meios para se coordenar em torno da construção do novo.

Neste artigo, apresentamos uma estrutura para a construção de índices reflexos, um novo tipo de ativo que ajudará outros sintéticos a florescer e estabelecerá um alicerce fundamental para todo o setor financeiro descentralizado.

Visão geral dos índices de

reflexo O objetivo de um índice de reflexo não é manter uma fixação específica, mas amortecer a volatilidade de sua garantia. Os índices permitem que qualquer pessoa ganhe exposição ao mercado de criptomoedas sem a mesma escala de risco que os ativos criptográficos reais. Acreditamos que o RAI, nosso primeiro índice de reflexo, terá utilidade imediata para outras equipes que emitem sintéticos no Ethereum (por exemplo, MakerDAO Multi-Collateral DAI [1], UMA [2], Synthetix [3]) porque dá aos seus sistemas uma menor exposição a ativos voláteis, como ETH, e oferece aos usuários mais tempo para sair de suas posições no caso de uma mudança significativa no mercado.

Para entender os índices reflexos, podemos comparar o comportamento de seu preço de resgate ao do preço de uma moeda estável.

O preço de resgate é o valor de uma unidade de dívida (ou moeda) no sistema. Destina-se a ser usado apenas como uma ferramenta de contabilidade interna e é diferente do preço de mercado (o valor pelo qual o mercado está negociando a moeda). No caso de stablecoins fiduciários, como USDC, os operadores do sistema declaram que qualquer pessoa pode resgatar uma moeda por um dólar americano e, portanto, o preço de resgate dessas moedas é sempre um. Existem também casos de stablecoins com base em criptografia, como o Multi Collateral DAI (MCD) da MakerDAO, em que o sistema tem como meta uma peg fixa de um dólar americano e, portanto, o preço de resgate também é fixado em um.

Na maioria dos casos, haverá uma diferença entre o preço de mercado de uma moeda estável e seu preço de resgate. Esses cenários criam oportunidades de arbitragem em que os comerciantes criarão mais moedas se o preço de mercado for superior ao de resgate e eles resgatarão suas moedas estáveis como garantia (por exemplo, dólares americanos no caso de USDC) caso o preço de mercado seja inferior ao preço de resgate.

Os índices de reflexo são semelhantes aos stablecoins porque eles também têm um preço de resgate que o sistema visa. A principal diferença no caso deles é que seu resgate não permanecerá fixo, mas é projetado para mudar enquanto é influenciado pelas forças do mercado. Na Seção 4, explicamos como o preço de resgate de um índice flutua e cria novas oportunidades de arbitragem para seus usuários.

Filosofia de design e estratégia de entrada no mercado

Nossa filosofia de design é priorizar a segurança, estabilidade e velocidade de entrega.

O Multi-Collateral DAI foi o lugar natural para começar a iterar no design da RAI. O sistema foi fortemente auditado e formalmente verificado, tem dependências externas mínimas e reuniu uma comunidade ativa de especialistas. Para minimizar o esforço de desenvolvimento e comunicação, queremos fazer apenas as alterações mais simples na base de código MCD original para alcançar nossa implementação.

Nossas modificações mais importantes incluem a adição de um definidor de taxa autônomo, um Oracle Network Medianizer que é integrado a muitos feeds de preços independentes e uma camada de minimização de governança destinada a isolar o

sistema o máximo possível da intervenção humana.

A primeira versão do protocolo (Estágio 1) incluirá apenas o definidor de taxa e outras pequenas melhorias na arquitetura central. Assim que provarmos que o setter funciona conforme o esperado, podemos adicionar com mais segurança o medianizador oracle (Estágio 2) e a camada de minimização de governança (Estágio 3).

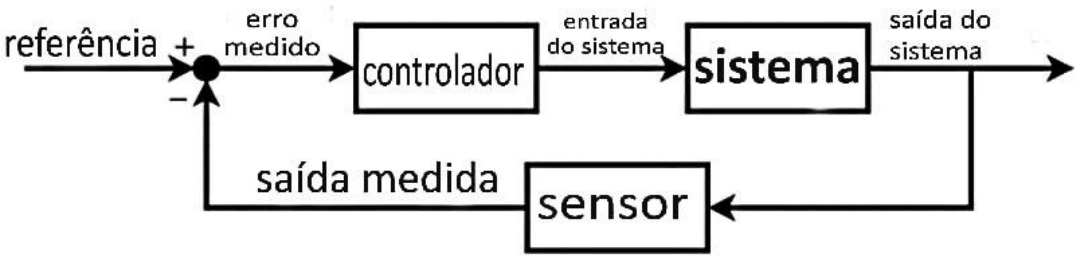
Mecanismos de política monetária

Introdução à Teoria de Controle

Um sistema de controle comum com o qual a maioria das pessoas está familiarizada é o chuveiro. Quando alguém começa a tomar banho, eles têm uma temperatura de água desejada em mente que, em teoria de controle, é chamado de *ponto de ajuste de referência*. A pessoa, atuando como o *controlador*, mede continuamente a temperatura do fluxo de água (que é chamada de sistema *saída*) e modifica a velocidade na qual eles giram o botão do chuveiro com base no *desvio* (ou *erro*) entre a temperatura desejada e a atual. A velocidade em a qual o botão é girado é chamada dedo sistema *entrada.* O objetivo é girar o botão rápido o suficiente para alcançar o ponto de ajuste de referência rapidamente, mas não tão rápido que o temperatura *excessiva*. Se houversistema *choques no* onde o fluxo de água a temperatura muda repentinamente, a pessoa deve ser capaz de manter a corrente temperatura sabendo com que rapidez girar o botão em resposta à perturbação.

A disciplina científica de manutenção da estabilidade em sistemas dinâmicos é chamada de controle teoria e encontrou ampla aplicação no controle de cruzeiro para carros, navegação aérea, reatores químicos, braços robóticos e processos industriais de todos os tipos. O bitcoin algoritmo de ajuste de dificuldade que mantém o tempo médio de bloqueio de dez minutos, apesar de um hashrate variável, é um exemplo de sistema de controle de missão crítica.

Na maioria dos sistemas de controle modernos, um *controlador algorítmico* é tipicamente embutido em o processo e recebe controle sobre uma entrada do sistema (por exemplo, o pedal do acelerador de um carro) em ordem para atualizá-lo automaticamente com base nos desvios entre a saída do sistema (por exemplo, um velocidade do carro) e o ponto de ajuste (por exemplo, a velocidade do controle de cruzeiro).



O tipo mais comum de controlador algorítmico é o *controlador PID*. Mais de 95% de aplicações industriais e uma ampla gama de sistemas biológicos empregam elementos dePID
controle[4]. Um controlador PID usa uma fórmula matemática com três partes para determinar sua saída:

C= ontroller saída proporcional Termo + termo integral + termo derivativo

O Termo Proporcional é a parte do controlador que é diretamente *proporcional* ao o desvio. Se o desvio for grande e positivo (por exemplo, a velocidade do controle de cruzeiro ponto de ajuste é muito maior do que a velocidade atual do carro), a resposta proporcional será grande e positivo (por exemplo, acelere).

O Termo Integral é a parte do controlador que leva em consideração por quanto tempo um o desvio persistiu. É determinado tomando a *integral* do desvio ao longo tempo e é usado principalmente para eliminar o *erro de estado estacionário*. Ele se acumula em ordem para responder a pequenos, embora persistentes desvios do ponto de ajuste (por exemplo, o cruzeiro o ponto de ajuste de controle foi 1 mph mais alto do que a velocidade do carro por alguns minutos).

O Termo Derivado é a parte do controlador que leva em consideração a rapidez o desvio está aumentando ou diminuindo. É determinado tomando a *derivada* do desvio

e serve para acelerar a resposta do controlador quando o desvio é crescendo (por exemplo, acelerar se o ponto de ajuste do controle de cruzeiro for maior do que a velocidade do carro e o carro começa a desacelerar). Também ajuda a reduzir o overshoot, desacelerando o resposta do controlador quando o desvio está diminuindo (por exemplo, aliviar o gás como o velocidade do carro começa a se aproximar do ponto de ajuste do controle de cruzeiro).

A combinação dessas três partes, cada uma das quais pode ser ajustada de forma independente, dá aos controladores PID grande flexibilidade no gerenciamento de uma ampla variedade de sistemas de controle formulários.

Os controladores PID funcionam melhor em sistemas que permitem algum grau de atraso na resposta tempo, bem como a possibilidade de ultrapassagem e oscilação em torno do ponto de ajuste como o sistema tenta se estabilizar. Sistemas de índice de reflexo como RAI são adequados para este tipo de cenário onde seus preços de resgate podem ser alterados por PID controladores.

De modo mais geral, foi descoberto recentemente que muitos dos atuais as regras de política monetária dos bancos (por exemplo, a Regra de Taylor) são, na verdade, aproximações dosPID
controladores[5].

Mecanismo de feedback da taxa de resgate

O mecanismo de feedback da taxa de resgate é o componente do sistema responsável por alterar o preço de resgate de um índice de reflexo. Para entender como funciona, nós primeiro precisa descrever por que o sistema precisa de um mecanismo de feedback em oposição a usando o controle manual e qual é a saída do mecanismo.

Componentes do mecanismo de feedback

Em teoria, seria possível manipular diretamente o índice de redenção do reflexo preço (descrito na Seção 2), a fim de influenciar os usuários do índice e, em última instância, alterar o preço de mercado do índice. Na prática, este método não teria o efeito desejado nos participantes do sistema. Da perspectiva de um titular SEGURO, se o resgate o preço é aumentado apenas uma vez, eles podem aceitar um preço mais alto por unidade de dívida, absorver a perda por um índice de garantia reduzido e manter

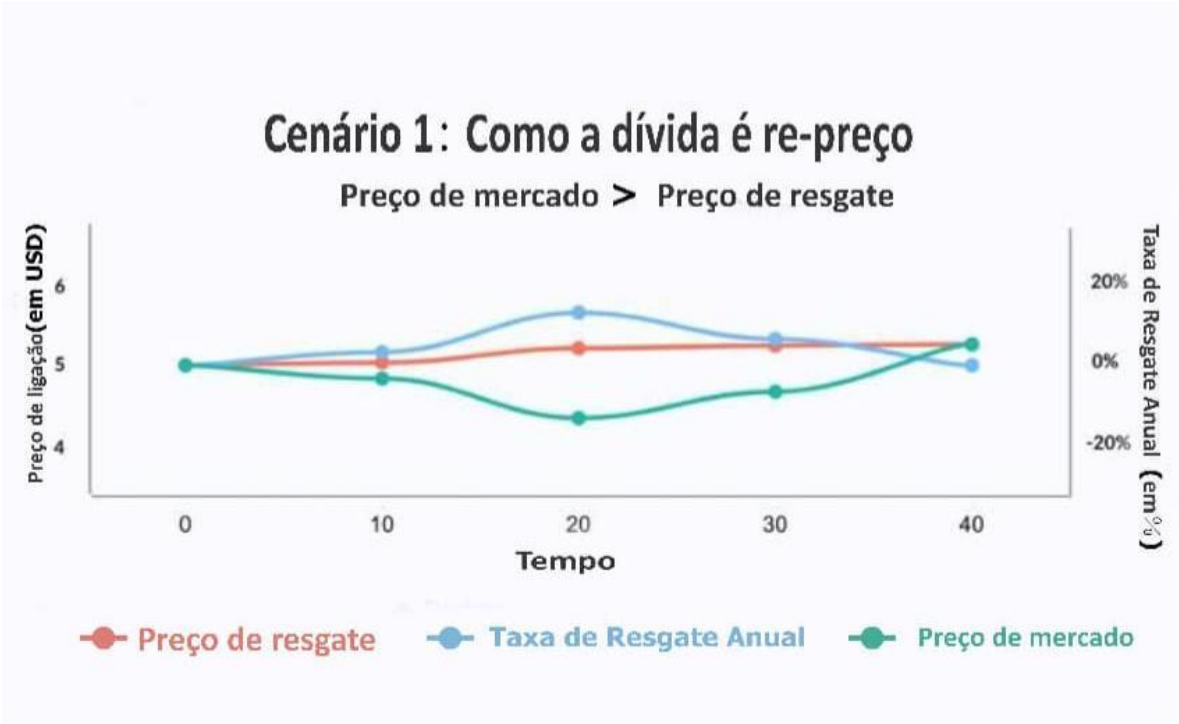
sua posição. Se, no entanto, eles esperam que o preço de resgate continue a aumentar ao longo do tempo, eles provavelmente estaria mais inclinado a evitar perdas futuras esperadas e, assim, escolher pagar pagar suas dívidas e fechar suas posições.

Esperamos que os participantes do sistema de índice de reflexo não respondam diretamente às mudanças no preço de resgate, mas em vez disso responda à *taxa de variação do preço de resgate* que chamamos de *taxa de resgate*. A taxa de resgate é definida por um *feedback mecanismo* que a governança pode ajustar ou permitir que seja totalmente automatizado.

Cenários de mecanismo de feedback

Lembre-se de que o mecanismo de feedback visa manter o equilíbrio entre os preço de resgate e o preço de mercado usando a taxa de resgate para contrabalançar mudanças nas forças de mercado. Para conseguir isso, a taxa de resgate é calculada de forma que opõe-se ao desvio entre os preços de mercado e de resgate.

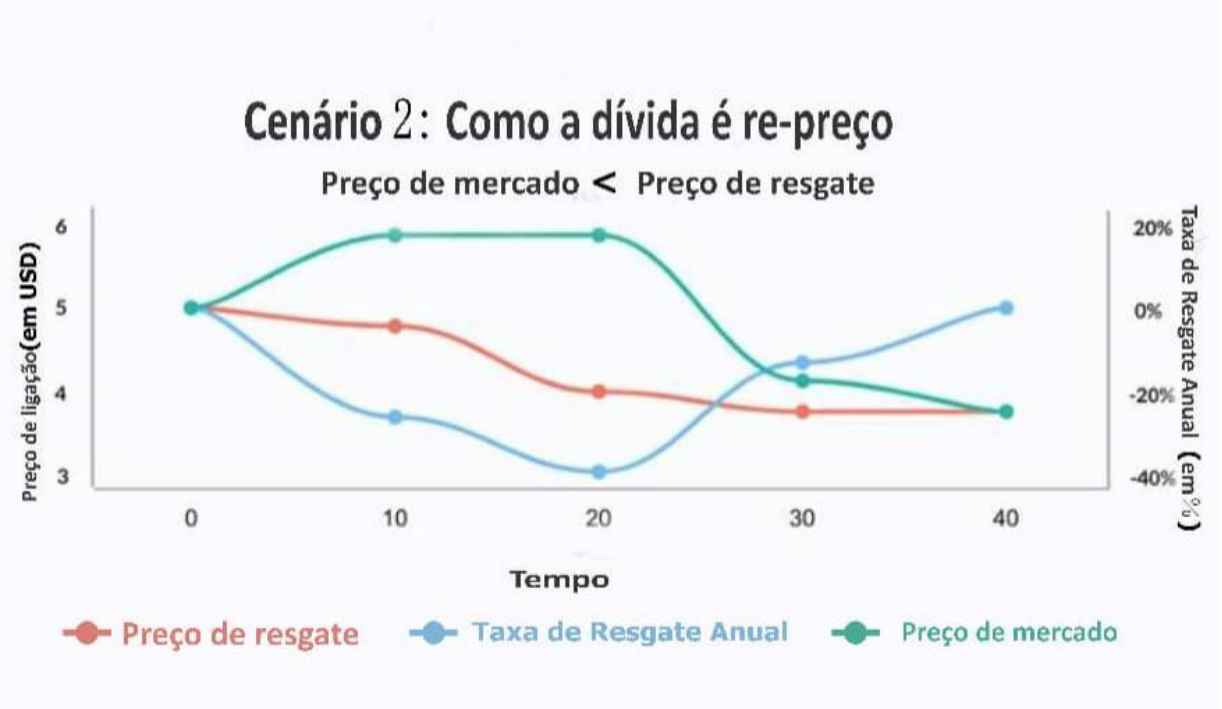
No primeiro cenário abaixo, se o preço de mercado do índice for superior ao seu resgate preço, o mecanismo irá calcular uma taxa negativa que começará a diminuir o preço de resgate, barateando a dívida do sistema.



A expectativa de uma redução no preço de resgate provavelmente desencorajará as pessoas de manter índices e encorajar os detentores do SAFE a gerar mais dívidas (mesmo que o o preço da garantia não muda) que é então vendido no mercado, equilibrando assim oferta e demanda. Observe que este é o cenário ideal onde os detentores de índice reaja rapidamente em resposta ao mecanismo de feedback. Na prática (e especialmente em nos primeiros dias após o lançamento), esperamos um atraso entre o início do mecanismo e resultados reais observados na quantidade de dívida emitida e posteriormente no mercado preço.

Por outro lado, no cenário dois, se o preço de mercado do índice for inferior ao preço de resgate, a taxa torna-se positiva e passa a repactuar toda a dívida para que fica mais caro.

À medida que a dívida se torna mais cara, os índices de garantia de todos os SAFEs caem (assim, os criadores do SAFE são incentivados a pagar suas dívidas) e os usuários começam a acumular índices com a expectativa de que aumentem de valor.



Algoritmo de mecanismo de feedback

No cenário a seguir, assumimos que o protocolo usa uma integral proporcional controlador para calcular a taxa de resgate:

- O índice reflexo é lançado com um preço de resgate arbitrário 'rand'

- Em algum ponto, o preço de mercado do índice sobe de 'rand' para 'rand' + x. Depois de o mecanismo de feedback lê o novo preço de mercado, calcula um termo proporcional *p*, que neste caso é -1 * (('rand' + x) / 'rand'). O proporcional é negativo para diminuir o preço de resgate e por sua vez repare os índices para que fiquem mais baratos

- Depois de calcular o proporcional, o mecanismo irá determinar o integral termo *i* adicionando todos os desvios anteriores dos últimos *deviationInterval* segundos de

- O mecanismo soma o proporcional e o integral e calcula um taxa de resgate por segundo *r* que lentamente começa a diminuir o resgate preço. Conforme os criadores do SAFE percebem que podem gerar mais dívidas, eles inundarão o mercado com mais índices
- Após *n* segundos, o mecanismo detecta que o desvio entre o os preços de mercado e de resgate são insignificantes (sob um parâmetro especificado *ruído*). Neste ponto, o algoritmo define r como zero e mantém o resgate preço onde está

Na prática, o algoritmo será mais robusto e faremos algumas variáveis imutável (por exemplo, o *ruído* parâmetro de, *deviationInterval*) ou haverá limites estritos sobre o que a governança pode mudar.

Ajuste do mecanismo de feedback

De extrema importância para o funcionamento adequado do sistema de índice de reflexo é o ajuste dos parâmetros do controlador algorítmico. Parametrização inadequada pode resultar no sistema sendo muito lento para alcançar estabilidade, ultrapassagem maciça, ou sendo geralmente instável diante de choques externos.

O processo de ajuste para um controlador PID normalmente envolve a execução do sistema ao vivo, ajustar os parâmetros de ajuste e observar a resposta do sistema, muitas vezes propositalmente introduzindo choques ao longo do caminho. Dada a dificuldade e risco financeiro de ajustar os parâmetros de um sistema de índice de reflexo ao vivo, planejamos alavancar modelagem e simulação de computador, tanto

quanto possível, para definir os parâmetros iniciais, mas também permitirá que a governança atualize os parâmetros de ajuste se dados adicionais da produção mostra que eles estão abaixo do ideal.

Configurador do mercado monetário

No RAI, pretendemos manter a taxa de empréstimo (taxa de juros aplicada ao gerar índices) fixos ou limitados e apenas modificam o preço de resgate, minimizando assim o complexidade envolvida na modelagem do mecanismo de feedback. A taxa de empréstimo em nosso caso é igual ao spread entre a taxa de estabilidade e DSR em Multi-Collateral DAI.

Embora planejemos manter a taxa de empréstimo fixa, é possível alterá-la ao lado do preço de resgate usando um configurador do mercado monetário. O mercado de dinheiro altera a taxa de empréstimo e o preço de resgate de uma forma que incentiva o SEGURO criadores para gerar mais ou menos dívidas. Se o preço de mercado de um índice estiver acima resgate, ambas as taxas começarão a diminuir, enquanto se estiver abaixo do resgate, as
taxas irão aumentar.

Acordo Global

A liquidação global é um método de último recurso usado para garantir o preço de resgate a todos os detentores de índice reflexo. Destina-se a permitir que os detentores de índice reflexo e SEGURO criadores para resgatar a garantia do sistema em seu valor líquido (quantidade de índices por cada tipo de garantia, de acordo com o último preço de resgate). Qualquer um pode acionar liquidação após queimar uma certa quantidade de tokens de protocolo.

A liquidação tem três fases principais:

- **Gatilho**: liquidação é acionada, os usuários não podem mais criar SAFEs, todos feeds de preço de garantia e o preço de resgate são congelados e registrados

- **Processo**: processar todos os leilões pendentes

- **Reivindicação**: cada detentor de índice de reflexo e criador SAFE pode

reivindicar uma quantidade fixa de qualquer garantia do sistema com base no último preço de resgate registrado do índice

Governança

A grande maioria dos parâmetros será imutável e a mecânica interna do contrato inteligente não será atualizável, a menos que os detentores de tokens de governança implantem um sistema inteiramente novo. Escolhemos essa estratégia porque podemos eliminar o meta-jogo em que as pessoas tentam influenciar o processo de governança em seu próprio benefício, prejudicando a confiança no sistema. Estabelecemos a operação adequada do protocolo sem colocar muita fé em humanos (o "efeito bitcoin") para maximizar a escalabilidade social e minimizar os riscos para outros desenvolvedores que queiram usar RAI como infraestrutura central em seus próprios projetos.

Para os poucos parâmetros que podem ser alterados, propomos a adição de um Módulo de Governança Restrita destinado a atrasar ou limitar todas as modificações possíveis do sistema. Além disso, apresentamos o Governance Ice Age, um registro de permissões que pode bloquear algumas partes do sistema do controle externo após o término de certos prazos.

Governança limitada pelo tempo A governança limitada pelo

tempo é o primeiro componente do Módulo de governança restrita. Ele impõe atrasos de tempo entre as alterações aplicadas ao mesmo parâmetro. Um exemplo é a possibilidade de alterar os endereços dos oráculos usados no Oracle Network Medianizer (Seção 6.2) após pelo menos *T* segundos terem passado desde a última modificação do oráculo.

Governança Limitada por Ações

O segundo componente do Módulo de Governança Restrita é a Governança Limitada por Ações. Cada parâmetro governável tem limites para quais valores ele pode ser definido e quanto ele pode mudar em um determinado período de tempo. Exemplos notáveis são as versões iniciais do mecanismo de feedback da taxa de resgate (Seção 4.2), que os detentores de tokens de governança poderão ajustar.

Governança da Idade do GeloIdade do

AGelo é um contrato inteligente imutável que impõe prazos para a alteração de parâmetros específicos do sistema e para a atualização do protocolo. Ele pode ser usado no caso em que a governança deseja ter certeza de que pode corrigir os bugs antes que o protocolo se bloqueie e negue a intervenção externa. A Idade do Gelo verificará se uma alteração é permitida comparando o nome do parâmetro e o endereço do contrato afetado em um registro de prazos. Se o prazo expirou, a chamada será revertida.

A governança pode atrasar a Idade do Gelo um número fixo de vezes se os bugs forem encontrados perto da data em que o protocolo deve começar a se bloquear. Por exemplo, a Era do Gelo só pode ser adiada três vezes, cada vez por um mês, para que as correções de bug recém-implementadas sejam testadas corretamente.

Áreas centrais onde a governança é necessária

Prevemos quatro áreas onde a governança pode ser necessária, especialmente nas primeiras versões desta estrutura:

- **Adicionar novos tipos de garantias**: o RAI será respaldado apenas pela ETH, mas outros índices serão respaldados por vários tipos de garantias e a governança será capaz
  de diversificar o risco ao longo do tempo

- **Mudança de dependências externas**: oráculos e DEXs dos quais o sistema depende podem ser atualizados. A governança pode apontar o sistema para dependências mais novas para que continue funcionando corretamente

- **Definidores de taxas de ajuste fino**: os primeiros controladores de política monetária terão parâmetros que podem ser alterados dentro de limites razoáveis (conforme descrito por Ação e Governança com limite de tempo)

- **Migração entre versões do sistema**: em alguns casos, a governança pode implantar um novo sistema, dar-lhe permissão para imprimir tokens de protocolo e retirar essa permissão de um sistema antigo. Esta migração é realizada com a ajuda do Módulo de Migração Restrita descrito abaixo

Módulo de Migração Restrita

O seguinte é um mecanismo simples para migrar entre as versões do sistema:

- Há um registro de migração que mantém o controle de quantos sistemas diferentes o mesmo token de protocolo cobre e quais sistemas podem ter a permissão negada para imprimir tokens de protocolo em um leilão de dívida

- Cada vez que a governança implanta uma nova versão do sistema, eles enviam o endereço do contrato de leilão de dívida do sistema no registro de migração. A governança também precisa especificar se algum dia será capaz de impedir o sistema de imprimir tokens de protocolo. Além disso, a governança pode, a qualquer momento, dizer que um sistema sempre será capaz de imprimir tokens e, portanto, nunca será migrado

- Há um período de espera entre a proposta de um novo sistema e a retirada das permissões de um antigo

- Um contrato opcional pode ser configurado para desligar automaticamente um sistema antigo depois de ter as permissões de impressão negadas.

O módulo de migração pode ser combinado com uma Idade do Gelo que dá automaticamente a sistemas específicos a permissão para sempre poderem imprimir tokens.

Desligamento automático do sistema

Há casos em que o sistema pode detectar automaticamente e, como resultado, disparar liquidação por si só, sem a necessidade de queimar tokens de protocolo:

- **Atrasos graves no feed de preços**: o sistema detecta que um ou mais dos feeds de preços de garantia ou índice não foram atualizados por um longo tempo

- **Migração do sistema**: este é um contrato opcional que pode encerrar o protocolo após um período de resfriamento passar do momento em que a governança

retira a capacidade do mecanismo de leilão de dívida de imprimir tokens de protocolo (Módulo de migração restrita, Seção 5.4.1)

- **Desvio consistente do preço de mercado**: o sistema detecta que o preço de mercado do índice foi *x%* desviadopor um longo tempo em comparação com apreço de resgate

Governança doserá capaz de atualizar esses módulos de desligamento autônomo enquanto ainda sendo limitado ou até a Idade do Gelo começar a bloquear algumas partes do sistema.

Oráculos

Existem três tipos de ativos principais que o sistema precisa para ler feeds de preços para: o índice, o token de protocolo e cada tipo de garantia na lista de permissões. Os feeds de preços podem ser fornecida por oráculos liderados pela governança ou por redes oráculos já estabelecidas.

Oráculos liderados pela governança

Os detentores de tokens de governança ou a equipe principal que lançou o protocolo podem fazer parceria com outras entidades que reúnem vários feeds de preços fora da rede e, em seguida, enviam um único transação para um contrato inteligente que medianiza todos os pontos de dados.

Esta abordagem permite mais flexibilidade na atualização e alteração do oráculo infra-estrutura, embora venha à custa da falta de confiança.

Oracle Network Medianizermedianizer de

Umrede oracle é um contrato inteligente que lê preços de vários fontes que não são controladas diretamente pela governança (por exemplo, Uniswap V2 pool entre um tipo de colateral de índice e outros stablecoins) e, em seguida, medianiza todos os resultados. ONM funciona da seguinte forma:

- Nosso contrato rastreia as redes oracle permitidas que ele pode chamar para solicitar preços de garantia. O contrato é financiado por parte do excedente do sistema acumula (usando o Tesouro Excedente, Seção 11). Cada rede oráculo

aceita tokens específicos como pagamento para que nosso contrato também acompanhe o quantidade mínima e o tipo de tokens necessários para cada solicitação

- Para empurrar um novo feed de preço no sistema, todos os oráculos precisam ser chamado de antemão. Ao chamar um oráculo, o contrato primeiro troca alguns taxas de estabilidade com um dos tokens aceitos pelo oráculo. Depois que um oráculo é chamado, o contrato marca a chamada como “válida” ou “inválida”. Se uma chamada for inválida, o oráculo defeituoso específico não pode ser chamado novamente até que todos os outros sejam chamados e o contrato verifica se há uma maioria válida. Uma chamada oracle válida não deve reverter e deve recuperar um preço que foi postado na rede em algum momento os últimos *m* segundos. “Recuperar” significa coisas diferentes dependendo de cada tipo de oráculo:

    - Para oráculos baseados em pull, dos quais podemos obter um resultado imediatamente, nosso o contrato precisa pagar uma taxa e buscar diretamente o preço

    - Para oráculos baseados em push, nosso contrato paga a taxa, chama o oráculo e precisa esperar um período específico de tempo *n* antes de chamar o oráculo novamente a fim de obter o preço solicitado

- Cada resultado do oráculo é salvo em um array. Depois que cada oráculo na lista de permissões é chamado e se a matriz tem pontos de dados válidos suficientes para formar uma maioria (por exemplo, o contrato recebeu dados válidos de 3/5 oráculos), os resultados são classificados e o contrato escolhe a mediana

- Quer o contrato encontre a maioria ou não, a matriz com os resultados do oracle é cancelado e o contrato precisará aguardar *p* segundos antes de iniciar todo o processo tudo de novo

O Oracle Network Backup

Governance pode adicionar uma opção de backup oracle que começa a empurrar os preços do sistema se o medianizador não consegue encontrar a maioria das redes oráculos válidas várias vezes seguidas.

A opção de backup deve ser definida quando o medianizer é implantado, pois não pode ser mudou depois. Além disso, um contrato separado pode monitorar se o backup foi

vem substituindo o mecanismo de medianização há muito tempo e fecha automaticamente o protocolo.

Cofres

A fim de gerar índices, qualquer pessoa pode depositar e alavancar sua garantia criptográfica dentro de cofres. Enquanto um SAFE é aberto, ele continuará acumulando dívidas de acordo com a taxa de empréstimo da garantia depositada. À medida que o criador do SAFE paga sua dívida, ele poderá retirar mais e mais de sua garantia bloqueada.

Ciclo de vida do SAFE

Há quatro etapas principais necessárias para a criação de índices reflexos e, subsequentemente, o pagamento de uma dívida do SAFE:

- Depositar garantia no SAFE

  O usuário primeiro precisa criar um novo SAFE e depositar a garantia nele. ● Gerar índices apoiados pela garantia do SAFE

  O usuário especifica quantos índices deseja gerar. O sistema cria um montante igual de dívida que começa a acumular de acordo com a taxa de empréstimo da garantia.

- Saldar a dívida SAFE

  Quando o criador do SAFE deseja retirar sua garantia, ele deve pagar sua dívida inicial mais os juros acumulados.

- Retirar a garantia

  Depois que o usuário pagar parte ou a totalidade de sua dívida, ele terá permissão para retirar a garantia.

Liquidação SAFE

A fim de manter o sistema solvente e cobrir o valor de toda a dívida pendente, cada SAFE pode ser liquidada no caso de seu índice de garantia cair abaixo de um

determinado limite. Qualquer pessoa pode iniciar uma liquidação, caso em que o sistema confiscará as garantias do SAFE e as venderá em um *leilão de garantias*.

Seguro de liquidação

Em uma versão do sistema, os criadores do SAFE podem ter a opção de escolher um *gatilho* para quando seus SAFEs forem liquidados. Os gatilhos são contratos inteligentes que adicionam automaticamente mais garantias em um SEGURO e, potencialmente, salvam-no da liquidação. Exemplos de gatilhos são contratos que vendem posições curtas ou contratos que se comunicam com protocolos de seguro como o Nexus Mutual [6].

Outro método para proteger os SAFEs é a adição de dois limites de colateralização diferentes: *seguro* e *risco*. Os usuários do SAFE podem gerar dívidas até atingirem o limite seguro (que é mais alto do que o risco) e só serão liquidados quando a garantia do SAFE ficar abaixo do limite do risco.

Leilões de garantias

Para iniciar um leilão de garantias, o sistema precisa usar uma variável chamada *liquidationQuantity* para determinar o valor da dívida a ser coberto em cada leilão e o valor correspondente de garantias a serem vendidas. Uma *penalidade de liquidação* será aplicada a cada SAFE leiloado.

Collateral leilão Parâmetros

MINIMUMBID	montantemínima de moedas que precisam ser oferecido emum lance
desconto de	Discountem que a garantia está sendo vendido

lowerCollateralMedianDeviation	Max diminuir desvio limite que a mediana garantia pode ter em relação ao preço oráculo

upperCollateralMedianDeviation	Max desvio limite superior que o mediano garantia pode ter em relação ao preço do oracle
lowerSystemCoinMedianDeviation	Max desvio do limite inferior que a alimentação de moeda sistema preço oracle pode ter em relação ao preço do oracle moeda sistema
upperSystemCoinMedianDeviation	Max desvio limite superior que a mediana garantia pode ter em relação ao preço do oracle moeda sistema
minSystemCoinMedianDeviation	Min desvio para o resultado mediano da moeda do sistema em relação ao preço de resgate, a fim de levar em consideração a mediana.

Mecanismo de leilão de garantia

O leilão de desconto fixo é uma forma direta (em comparação com os leilões ingleses) de colocar garantias à venda em e xchange para moedas do sistema usadas para liquidar dívidas incobráveis. Os licitantes só precisam permitir que a casa de leilões transfira seu safeEngine.coinBalance e podem então chamar buyCollateral para trocar suas moedas do sistema por garantias que são vendidas com um desconto em relação ao último preço de mercado registrado.

Os licitantes também podem revisar o valor da garantia que podem obter de um leilão específico chamando getCollateralBought ou getApproximateCollateralBought. Observe que getCollateralBought não está marcado como view porque lê (e também atualiza) o redemptionPrice do relayer oracle, enquanto getApproximateCollateralBought usa lastReadRedemptionPrice.

Leilões de dívida

No cenário em que um leilão de garantia não pode cobrir todas as dívidas inadimplentes em um SEGURO e se o sistema não tiver reservas excedentes,

qualquer pessoa pode acionar um leilão de dívida.
Os leilões de dívida têm como objetivo cunhar mais tokens de protocolo (Seção 10) e vendê-los por índices que podem anular a inadimplência remanescente do sistema.

A fim de iniciar um leilão de dívida, as necessidades do sistema de usar dois parâmetros:

- initialDebtAuctionAmount:a quantidade inicial de fichas de protocolo para hortelã pós-leilão
- debtAuctionBidSize:o tamanho inicial de oferta (quantos índices devem ser oferecidas em troca de *initialDebtAuctionAmount* fichas de protocolo)

Definição de Parâmetros de Leilão de Dívida Autônoma

A quantidade inicial de tokens de protocolo cunhados em um leilão de dívida pode ser definida por meio de uma votação de governança ou pode ser ajustada automaticamente pelo sistema. Uma versão automatizada precisaria ser integrada aos oráculos (Seção 6) a partir dos quais o sistema leria o token de protocolo e os preços de mercado do índice de reflexo. The system would then set the initial amount of protocol tokens (*initialDebtAuctionAmount*) that will be minted for *debtAuctionBidSize* indexes. *initialDebtAuctionAmount* can be set at a discount compared to the actual PROTOCOL/INDEX market price in order to incentivize bidding.

Debt Auction Parameters

amountSoldIncrease	Increase in the amount of protocol tokens to be minted for the same amount of indexes
bidDecrease	Next bid's minimum decrease in the accepted amount of protocol tokens for the same amount of indexes
bidDuration	How long the bidding lasts after a new bid is submitted (in seconds)

totalAuctionLength	Total length of the auction (in seconds)
auctionsStarted	How many auctions have started until now

Debt Auction Mechanism

As opposed to collateral auctions, debt auctions only have one stage:

`decreaseSoldAmount(uint id, uint amountToBuy, uint bid)`: decrease the amount of protocol tokens accepted in exchange for a fixed amount of indexes.

The auction will be restarted if it has no bids placed. Every time it restarts, the system will offer more protocol tokens for the same amount of indexes. The new protocol token amount is calculated as *lastTokenAmount* * *amountSoldIncrease* / 100. After the auction settles, the system will mint tokens for the highest bidder.

Protocol Tokens

As described in earlier sections, each protocol will need to be protected by a token that is minted through debt auctions. Apart from protection, the token will be used to govern a few system components. Also, the protocol token supply will gradually be reduced with the use of surplus auctions. The amount of surplus that needs to accrue in the system before extra funds are auctioned is called the *surplusBuffer* and it is automatically adjusted as a percentage of the total debt issued.

Insurance Fund

Apart from the protocol token, governance can create an insurance fund that holds a wide array of uncorrelated assets and which can be used as a backstop for debt auctions.

Surplus Auctions

Surplus auctions sell stability fees accrued in the system for protocol tokens that are then burned.

Surplus Auction Parameters

bidIncrease	Minimum increase in the next bid
bidDuration	How long the auction lasts after a new bid is submitted (in seconds)
totalAuctionLength	Total length of the auction (in seconds)
auctionsStarted	How many auctions have started until now

Surplus Auction Mechanism

Surplus auctions have a single stage:

`increaseBidSize(uint id, uint amountToBuy, uint bid)`: anyone can bid a higher amount of protocol tokens for the same amount of indexes (surplus). Every new bid needs to be higher than or equal to *lastBid* * *bidIncrease* / 100. The auction will end after maximum *totalAuctionLength* seconds or after *bidDuration* seconds have passed since the latest bid and no new bids have been submitted in the meantime.

An auction will restart if it has no bids. On the other hand, if the auction has at least one bid, the system will offer the surplus to the highest bidder and will then burn all the gathered protocol tokens.

Surplus Indexes Management

Every time a user generates indexes and implicitly creates debt, the system starts applying a borrowing rate to the user's SAFE. The accrued interest is pooled in two different smart contracts:

- The *accounting engine* used to trigger debt (Section 9.2) and surplus (Section 10.1) auctions
- The *surplus treasury* used to fund core infrastructure components and incentivize external actors to maintain the system

The surplus treasury is in charge of funding three core system components:

- Oracle module (Section 6). Depending on how an oracle is structured, the treasury either pays governance whitelisted, off-chain oracles or it pays for calls toward oracle networks. The treasury can also be set up to pay the addresses that spent gas to call an oracle and update it

- In some cases, independent teams that maintain the system. Exemplos são teams who whitelist new collateral types or fine tune the system's rate setter (Seção 4.2)

The treasury can be set up so that some surplus recipients will automatically be denied funding in the future and others can take their place.

External Actors

The system depends on external actors in order to function properly. These actors are economically incentivized to participate in areas such as auctions, global settlement processing, market making and updating price feeds in order to maintain the system's health.

We will provide initial user interfaces and automated scripts to enable as many people as possible to keep the protocol secure.

Addressable Market

We see RAI as being useful in two main areas:

- **Portfolio diversification**: investors use RAI to get dampened exposure to an asset like ETH without the whole risk of actually holding ether

- **Collateral for synthetic assets**: RAI can offer protocols such as UMA, MakerDAO and Synthetix a lower exposure to the crypto market and give users more time

to exit their positions in the case of scenarios such as Black Thursday from March 2020 when millions of dollars worth of crypto assets were liquidated

Future Research

To push the boundaries of decentralized money and bring further innovation in decentralized finance, we will continue to look for alternatives in core areas such as governance minimization and liquidation mechanisms.

We first want to lay the groundwork for future standards around protocols that lock themselves from outside control and for true “money robots” which adapt in response to market forces. Afterwards, we invite the Ethereum community to debate and design improvements around our proposals with a specific focus on collateral and debt auctions.

Risks and Mitigation

There are several risks involved in developing and launching a reflex index, as well as subsequent systems that are built on top:

- **Smart contract bugs**: the greatest risk posed to the system is the possibility of a bug that allows anyone to extract all the collateral or locks the protocol in a state it cannot recover from. We plan to have our code reviewed by multiple security researchers and launch the system on a testnet before we commit to deploying it in production

- **Oracle failure**: we will aggregate feeds from multiple oracle networks and there will be strict rules in place for upgrading only one oracle at a time so that malicious governance cannot easily introduce false prices

- **Collateral black swan events**: there is the risk of a black swan event in the underlying collateral which can result in a high amount of liquidated SAFEs. Liquidations may not be able to cover the entire outstanding bad debt and so the system will continuously change its surplus buffer in order to cover a decent amount of issued debt and withstand market shocks

- **Improper rate setter parameters**: autonomous feedback mechanisms are highly experimental and may not behave exactly like we predict during simulations.

We plan to allow governance to fine-tune this component (while still being bounded) in order to avoid unexpected scenarios

- **Failure to bootstrap a healthy liquidator market**: liquidators are vital actors that make sure all issued debt is covered by collateral. We plan to create interfaces and automated scripts so that as many people as possible can participate in keeping the system secure.

Summary

We have proposed a protocol that progressively locks itself from human control and issues a low volatility, collateralized asset called a reflex index. We first presented the autonomous mechanism meant to influence the index's market price and then described how several smart contracts can limit the power that token holders have over the system. We outlined a self-sustaining scheme for medianizing price feeds from multiple independent oracle networks and then finished by presenting the general mechanism for minting indexes and liquidating SAFEs.

References

[1] “The Maker Protocol: MakerDAO's Multi Collateral Dai (MCD) System”, https://bit.ly/2YL5S6j

[2] “UMA: A Decentralized Financial Contract Platform”, https://bit.ly/2Wgx7E1

[3] Synthetix Litepaper, https://bit.ly/2SNHxZO

[4] KJ Åström, RM Murray, “Feedback Systems: An Introduction for Scientists and Engineers”, https://bit.ly/3bHwnMC

[5] RJ Hawkins, JK Speakes, DE Hamilton, “Monetary Policy and PID Control”, https://bit.ly/2TeQZFO

[6] H. Karp, R. Melbardis, “A peer-to-peer discretionary mutual on the Ethereum blockchain”, https://bit.ly/3du8TMy

[7] H. Adams, N. Zinsmeister, D. Robinson, “Uniswap V2 Core”, https://bit.ly/3dqzNEU

Glossary

Reflex index: a collateralized asset that dampens the volatility of its underlying

RAI: our first reflex index

Redemption Price: the price that the system wants the index to have. It changes, influenced by a redemption rate (computed by RRFM), in case the market price is not close to it. Meant to influence SAFE creators to generate more or pay back some of their debt

Borrowing Rate: annual interest rate applied to all SAFEs that have outstanding debt

Redemption Rate Feedback Mechanism (RRFM): an autonomous mechanism which compares the market and redemption prices of a reflex index and then computes a redemption rate that slowly influences SAFE creators to generate more or less debt (and implicitly tries to minimize the market/redemption price deviation)

Money Market Setter (MMS): a mechanism similar to RRFM which pulls multiple monetary levers at once. In the case of reflex indexes, it modifies both the borrowing rate and the redemption price

Oracle Network Medianizer (ONM): a smart contract that pulls prices from multiple oracle networks (which are not controlled by governance) and medianizes them if a majority (eg 3 out of 5) returned a result without throwing

Restricted Governance Module (RGM): a set of smart contracts that bound the power that governance tokens holders have over the system. It either enforces time delays or limits the possibilities that governance has to set certain parameters

Governance Ice Age: immutable contract that locks most components of a protocol from outside intervention after a certain deadline has passed

Accounting Engine: system component which triggers debt and surplus auctions. It also keeps track of the amount of currently auctioned debt, unactioned bad debt and the surplus buffer

Surplus Buffer: amount of interest to accrue and keep in the system. Any interest accrued above this threshold gets sold in surplus auctions that burn protocol tokens

Surplus Treasury: contract that gives permission to different system modules to withdraw accrued interest (eg ONM for oracle calls)