Rai : un collatéral à faible volatilité et sans risque de contrepartie pour l'écosystème DeFi

Stefan C. Ionescu, Ameen Soleimani

Mai 2020

Résumé. Nous présentons un protocole décentralisé à gouvernance minimale, qui modifie la valeur cible de son actif natif en réagissant automatiquement aux forces du marché. Le protocole permet à chacun d'obtenir un effet de levier sur ses crypto-actifs en émettant un "index réflexe", un actif à la volatilité atténuée par rapport à son collatéral. Nous indiquons comment les index peuvent être utiles en tant que collatéraux universels faiblement volatiles capables de protéger leurs détenteurs et d'autres protocoles de finance décentralisée de mouvements de marché soudains. Nous présentons nos plans pour assister d'autres équipes à lancer leur propres produits dérivés s'appuyant sur notre infrastructure. Enfin, nous offrons des alternatives aux systèmes d'oracles et de gouvernance aujourd'hui répandus de nombreux protocoles DeFi.

Traduction par ben.oxmo & PhilH

Table des matières

1. Introduction

La monnaie est l'un des mécanismes de coordination les plus puissants sur lesquels l'humanité s'appuie pour prospérer. Le droit de battre monnaie a été un privilège historiquement réservé au pouvoir souverain et à l'élite financière, imposé à la masse sans qu'elle ait son mot à dire. Bitcoin a démontré le potentiel d'une approche alternative pour créer une réserve de valeur monétaire indépendante des Etats. Ethereum offre une plateforme de création d'instruments synthétiques adossés à d'autres actifs, qui peuvent être peu volatiles, utilisés comme collatéraux, indexés sur un prix de référence et servir de moyen d'échange pour les transactions quotidiennes, en s'appuyant sur les mêmes principes de consensus décentralisé.

La liberté absolue de stocker de la valeur avec Bitcoin et les instruments synthétiques décentralisés d'Ethereum sont les fondements de la révolution du système financier qui s'annonce.

Dans ce document, nous proposons une approche structurée de création d'index réflexes, un nouveau type d'actif qui favorisera le développement d'autres produits dérivés et constituera une brique de base essentielle pour l'ensemble de la finance décentralisée.

2. Présentation des index réflexes

Le but d'un index réflexe n'est pas de maintenir un prix à une valeur spécifique, mais d'amortir la volatilité de son collatéral. Les index réflexes permettent à chacun de bénéficier de l'exposition au marché des valeurs crypto sans souffrir du niveau élevé de risque associé à la possession de ces actifs. Nous pensons que RAI, notre premier index réflexe, offrira une utilité immédiate aux autres projets proposant des produits dérivés sur Ethereum (comme le DAI multi-collatéralisé de MakerDAO [1], UMA [2], Synthetix [3]), en réduisant leur exposition à la volatilité d'actifs comme ETH et en permettant à leurs utilisateurs d'avoir le temps de clore leurs positions en cas de mouvements importants sur les marchés.

Pour comprendre comment fonctionnent les index réflexes, on peut comparer leur mécanisme de rédemption à celui des stablecoins.

Le prix de rédemption est la valeur d'une unité de dette dans le système. Il s'agit d'un instrument de comptabilisation propre au système et différent de son prix de marché, qui est celui auquel on peut acheter et vendre l'actif. Dans le cas des stablecoins adossés aux monnaies fiduciaires tel que l'USDC, les opérateurs du système déclarent que toute personne peut demander l'échange d'un stablecoin contre un dollar américain. Le prix de rédemption, i.e. la valeur demandée lors de cet échange d'une unité de monnaie, est toujours de un pour un. Il existe également des stablecoins adossés à des crypto-monnaies telles que MakerDAO's MultiCollateral DAI (MCD) pour lesquels le système a pour objectif d'indexer son unité de monnaie sur la valeur d'un dollar. Son prix de rédemption est donc aussi égal à un.

La plupart du temps, il y a une différence entre le prix de marché du stablecoin et son prix de rédemption. Il s'ensuit des opportunités d'arbitrage où les traders créent plus d'unités du stablecoin si le prix du marché est supérieur à son prix de rédemption. Inversement, ils échangent des unités du stablecoin contre le collatéral, par exemple en dollars américains dans le cas de l'USDC, lorsque le prix du marché est inférieur à son prix de rédemption.

Les index réflexes sont similaires aux stablecoin dans le sens où ils ont également un prix de rédemption ciblé par le système. En revanche, ce prix de rédemption n'est pas fixé une fois pour toutes : il est conçu pour évoluer en fonction des les forces du marché. Dans la section 4, nous expliquons comment le prix de rédemption d'un index flotte et crée de nouvelles opportunités d'arbitrage pour ses utilisateurs.

3. Principes de conception et stratégie de mise sur le marché

Nous mettons au premier rang de nos priorités la sécurité, la stabilité et la vitesse d'exécution.

Le Multi-Collateral DAI (MCD) était un point de départ naturel pour commencer à itérer sur la conception de RAI. Le système a été audité à plusieurs reprises et vérifié formellement, il a peu de dépendances externes et il rassemble une communauté active d'experts. Afin de minimiser l'effort de développement et de communication, nous avons réalisé notre implémentation en effectuant des changements à la marge du code d'origine du MCD.

Les modifications les plus importantes comportent l'ajout d'un système autonome de fixation de taux, un calculateur de médianes d'oracles qui traite plusieurs flux de prix indépendants, et la minimisation de la gouvernance afin de protéger au maximum le système des interventions humaines.

La toute première version du protocole (phase 1) n'inclura que le système de fixation du taux et quelques améliorations mineures à l'architecture de base. Une fois le bon fonctionnement du taux validé, nous pourrons implémenter le calculateur de médianes d'oracles (phase 2) et la gouvernance minimale (phase 3).

4. Mécanismes de politique monétaire

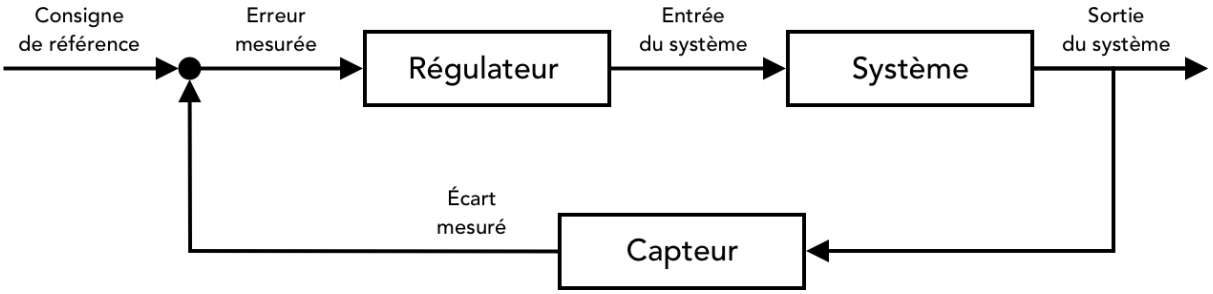
4.1. Introduction à la théorie du contrôle

Un système de contrôle commun que chacun connaît est la douche. Lorsque quelqu'un fait couler une douche, il a en tête une température d'eau désirée qui, dans la théorie du contrôle, est appelée la *consigne de référence*. La personne, agissant en tant que *contrôleur*, mesure en continu la température de l'eau (c'est *la sortie* du système) et tourne plus ou moins vite le robinet de la douche en fonction de *l'écart* (ou erreur) entre la température souhaitée et la température constatée. La vitesse à laquelle le robinet est tourné est *l'entrée* du système. L'objectif est donc de tourner assez rapidement le robinet pour obtenir la consigne de référence, sans être trop brusque afin d'éviter que la

température ne devienne excessivement froide ou chaude. Si le système subit des à-coups conduisant à des changements soudains de la température de l'eau, la personne devrait pouvoir maintenir la température visée en anticipant la vitesse à laquelle il faut tourner le robinet pour réduire ces à-coups.

La discipline scientifique du maintien de la stabilité dans les systèmes dynamiques est appelée la théorie du contrôle. Cette théorie est appliquée dans de nombreux domaines industriels, tels que la conduite de véhicule assistée ou autonome, les réacteurs chimiques, les bras robotiques, ou encore l'aéronautique. L'algorithme d'ajustement de la difficulté pour le Bitcoin qui maintient le pas de temps de dix minutes pour la validation d'un bloc, et ce en dépit d'un taux de hachage variable, est un exemple de mécanisme de contrôle jouant un rôle critique.

Dans la plupart des systèmes de contrôle modernes, un *contrôleur algorithmique* est intégré dans le processus et on lui donne le contrôle d'une entrée du système (par exemple, la pédale d'accélérateur d'une voiture) afin de la modifier automatiquement en fonction de l'écart entre la sortie du système (la vitesse de la voiture) et le point de consigne (la vitesse cible assignée au régulateur de vitesse).



Le type de contrôleur algorithmique le plus courant est le *régulateur PID*. Plus de 95% des applications industrielles et une large partie des systèmes biologiques utilisent des éléments de *régulation PID* [4].

Un régulateur PID utilise une formule mathématique en trois parties pour déterminer sa sortie :

Sortie du régulateur = terme proportionnel + terme intégral + terme dérivé

Le *terme proportionnel* est la partie du régulateur qui est directement proportionnelle à l'écart. Si l'écart est important et positif (par exemple, la vitesse de consigne du régulateur de vitesse est bien plus élevée que la vitesse actuelle de la voiture) la réponse proportionnelle sera aussi importante et positive (i.e. appuyer au maximum sur la pédale d'accélérateur).

Le *terme intégral* est la partie du régulateur qui prend en compte la durée de la déviation du système. Il est déterminé en prenant *l'intégrale* de la déviation dans le temps afin d'éliminer les erreurs en régime stationnaire. Il s'accumule avec l'objectif de répondre à de petits écarts persistants par rapport au point de consigne (par ex. la consigne de contrôle est supérieure de 1 km/h à la vitesse de la voiture pendant quelques minutes).

Le *terme dérivé* est la partie du régulateur qui prend en compte la vitesse à laquelle l'écart augmente ou diminue. Il est déterminé en prenant la *dérivée* de l'écart et sert à accélérer la réponse du régulateur lorsque l'écart est croissant (par exemple, accélérer si la consigne du régulateur de vitesse est supérieure à la vitesse de la voiture alors que la voiture commence à ralentir). Il aide également à amortir la réponse du régulateur lorsque l'écart décroît (par ex. décélérer quand la vitesse de la voiture commence à se rapprocher de la consigne du régulateur de vitesse).

La combinaison de ces trois composantes, dont chacune peut être réglée indépendamment, donne aux régulateurs PID une grande flexibilité pour gérer une large variété de systèmes de contrôle.

Les régulateurs PID fonctionnent de façon optimale dans les systèmes qui tolèrent une certaine latence du temps de réponse, ainsi que des à-coups et des oscillations à l'approche de la consigne visée. Les systèmes d'index réflexes comme RAI sont bien adaptés à ce type de scénario en soumettant la correction des prix de rédemption à l'action de régulateurs PID.

Plus généralement, il a été récemment découvert que bon nombre des règles de politique monétaire des banques centrales (par exemple la règle de Taylor) reproduisent de façon approximative le fonctionnement de régulateurs PID [5].

4.2. Mécanisme de rétroaction sur le taux de rédemption

Le mécanisme de rétroaction sur le taux de rédemption est le composant du système en charge de la modification du prix de rédemption de l'index réflexe. Pour comprendre son fonctionnement, il nous faut d'abord expliquer pourquoi un mécanisme de rétroaction est préférable à un contrôle manuel, et quelles sont les données qu'il produit.

4.2.1. Composants du mécanisme de rétroaction

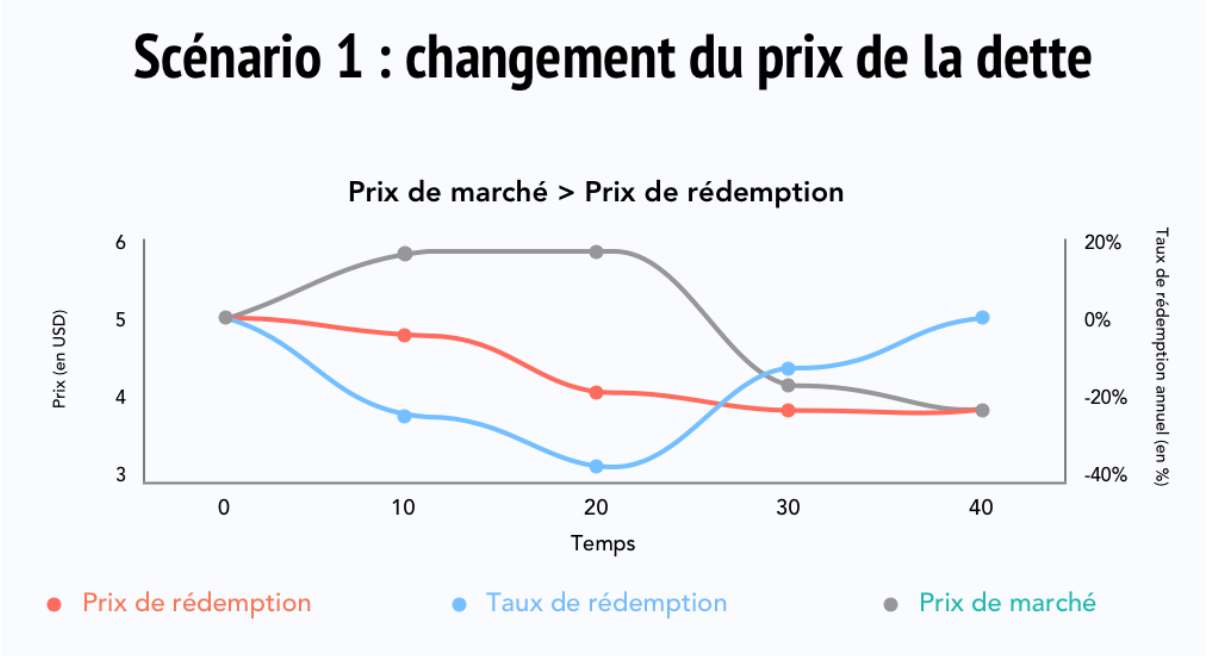
En théorie, il est possible de modifier directement le prix de rédemption de l'index réflexe (décrit en section 2) afin d'influer sur les utilisateurs et au bout du compte sur son prix de marché. En pratique, cette méthode n'aurait pas l'effet espéré sur les participants. Si le prix de rédemption était modifié une seule fois, un créateur de SAFE pourrait accepter un prix plus élevé par unité de dette, et maintenir sa position malgré la perte entraînée par la baisse du ratio de collatéralisation. Mais si l'on peut s'attendre à ce que le prix de rédemption continue à augmenter au fil du temps, il serait plus avantageux d'éviter des pertes à venir, et donc de rembourser la dette et clôturer ses positions.

Nous nous attendons à ce que les participants au système d'index réflexes ne répondent pas directement aux changements du prix de rédemption, mais au *taux de changement du prix de rédemption*, que nous appelons le *taux de rédemption*. Le taux de rédemption est déterminé par un *mécanisme de rétroaction* que la gouvernance peut ajuster, ou qui peut être totalement automatisé.

4.2.2. Scénarios de fonctionnement du mécanisme de rétroaction

Rappelons que le mécanisme de rétroaction vise à maintenir l'équilibre entre le prix de rédemption et le prix de marché en s'appuyant sur le taux de rédemption pour contrer les fluctuations du marché. Pour atteindre cet objectif, le taux de rédemption est calculé de manière à neutraliser les divergences entre les prix de rédemption et de marché.

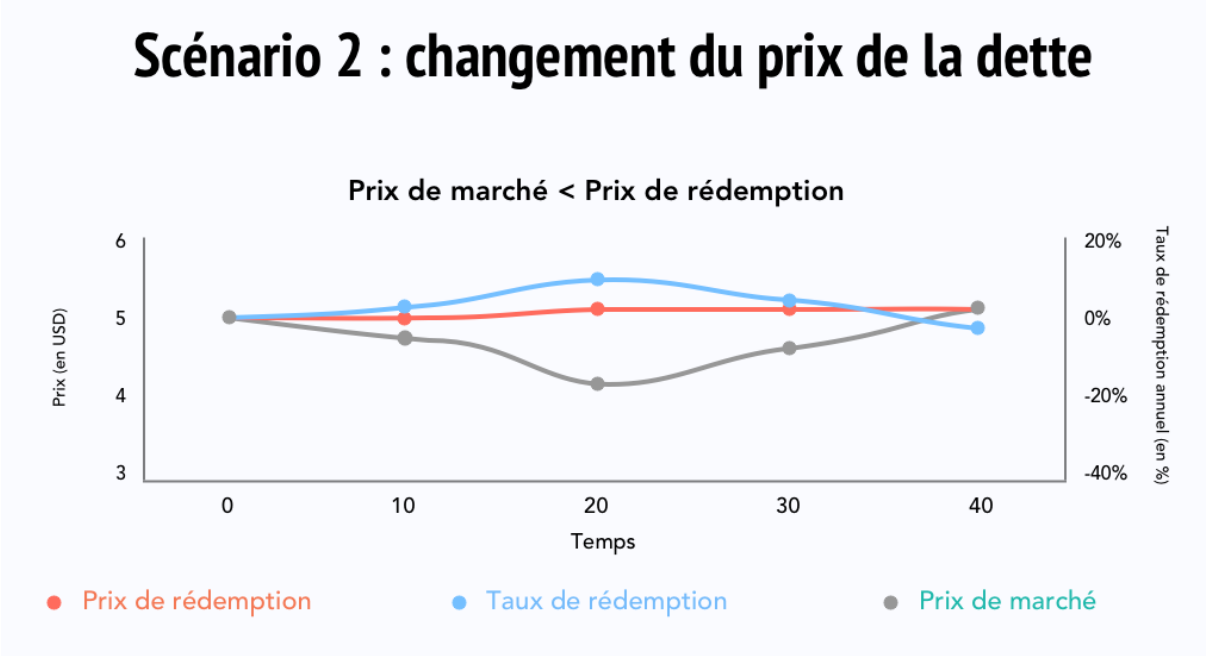
Dans le premier scénario ci-dessous, si le prix de marché de l'index est supérieur à son prix de rédemption, le mécanisme produit un taux négatif qui initie une baisse du prix de rédemption, ce qui a pour effet collatéral de faire décroitre la dette pour les créateurs de SAFE.



La perspective d'une baisse du prix de rédemption aura pour effet probable de décourager la conservation des index et d'encourager les créateurs de SAFE à produire plus de dette (même si le prix du collatéral ne change pas), qui sera ensuite vendue sur le marché et conduira à équilibrer demande et offre. Notons qu'il s'agit ici d'un scénario idéal où les détenteurs d'index réagissent rapidement au mécanisme de rétroaction. En pratique (et notamment pendant la période initiale, consécutive au lancement), nous nous attendons à ce qu'il y ait une latence entre le déclenchement du mécanisme et les résultats attendus en termes de quantité de dette et de prix de marché.

A l'inverse, si le prix de marché de l'index devient inférieur au prix de rédemption, le taux de rédemption devient positif et commence à impacter à la hausse la dette qui devient plus coûteuse.

Lorsque la dette devient plus coûteuse, le ratio de collatéralisation des SAFE baisse (les créateurs de SAFE sont ainsi poussés à rembourser leur dette) et les utilisateurs commencent à accumuler les index dans la perspective de leur appréciation à venir.



4.2.3. Algorithme du mécanisme de rétroaction

Dans le scénario suivant, nous faisons l'hypothèse que le protocole utilises un régulateur proportionnel et intégral pour calculer le taux de rédemption :

- L'index réflexe est lancé avec un prix de rédemption arbitraire 'rand'
- A un moment donné, le prix de marché de l'index augmente, passant de 'rand' à 'rand' + x. Après que le mécanisme de rétroaction ait pris connaissance du nouveau prix de marché, il calcule le terme proportionnel *p*, qui dans ce cas correspond à -1 * (('rand' + x) / 'rand'). La formule donne un résultat négatif afin de faire décroître le prix de rédemption et par conséquent, faire baisser le prix des index
- Après avoir calculé le terme proportionnel, le mécanisme détermine l'intégrale *i* en ajoutant toutes les déviations passées depuis les dernières *deviationInterval* secondes
- Le mécanisme fait la somme de la proportionnelle et de l'intégrale et calcule un taux de rédemption par seconde *r* qui commence lentement à faire décroître le prix de rédemption. Lorsque les créateurs de SAFE réalisent qu'ils peuvent générer plus de dette, ils mettent sur le marché de nouveaux index
- After *n* secondes, le mécanisme détecte que la déviation entre les prix de marché et de rédemption est négligeable (compte tenu d'une marge d'erreur prédéfinie *noise*). A ce stade, l'algorithme réduit *r* à zéro et maintient le prix de rédemption à sa valeur courante

Au fil du temps, l'algorithme démontrera sa robustesse et certaines variables comme le paramètre *noise* ou *deviationInterval* seront figées ou encadrées entre des bornes que la gouvernance ne pourra plus étendre.

4.2.4. Réglage du mécanisme de rétroaction

Le réglage des paramètres du régulateur algorithmique est essentiel au bon fonctionnement du système d'index réflexes. Des paramètres inadéquats pourraient conduire à un système trop lent pour atteindre la stabilité, ou à l'inverse ayant des réactions trop brutales, et de façon générale se comportant de façon instable lors de chocs externes.

Le processus de réglage pour un régulateur PID implique de faire tourner le système, d'ajuster les paramètres et d'observer les réponses du système, notamment lors de chocs introduits à dessein. Etant donné la difficulté et le risque financier d'un ajustement des paramètres d'un index réflexe en production, nous prévoyons de modéliser et de simuler le fonctionnement du système pour fixer les paramètres initiaux, et de permettre à la gouvernance de les modifier ultérieurement si les données de production montrent que c'est nécessaire.

4.3. Configurateur de taux du marché monétaire

Avec le RAI, nous prévoyons de maintenir le taux d'emprunt (taux d'intérêt appliqué lors de la génération des index) fixe ou plafonné, et de modifier uniquement le prix de rédemption, afin de limiter la complexité du modèle du mécanisme de rétroaction. Le taux d'intérêt dans notre cas est égal à la marge entre les frais de stabilité et le DSR (DAI Savings Rate) dans le DAI multi-collatéralisé.

Même si nous prévoyons de maintenir le taux d'emprunt fixe, il est possible de le changer en même temps que le prix de rédemption en utilisant un marché monétaire. Le marché monétaire modifie le taux d'emprunt et le prix de rédemption afin d'inciter les créateurs de SAFE à générer plus ou moins de dette. Si le prix de marché d'un index est au-dessus du prix de rédemption, les deux taux commenceront à baisser ; s'il est en-dessous, ils augmenteront.

4.4. Règlement global

Le règlement global est une méthode de dernier recours utilisée pour garantir le prix de rédemption à tous les détenteurs d'index réflexes. Il est destiné à permettre à la fois aux détenteurs d'index réflexes et aux créateurs de SAFE de racheter le collatéral du système à sa valeur nette (montant des index pour chaque type de collatéral, selon le dernier prix de rédemption). Tout le monde peut déclencher un règlement en détruisant une certaine quantité de jetons de protocole.

Le règlement comporte trois phases principales:

- **Déclencheur** : le règlement est déclenché, les utilisateurs ne peuvent plus créer de SAFE, tous les flux de prix des collatéraux et le prix de rédemption sont gelés et enregistrés
- **Processus** : toutes les enchères en cours sont traitées

- **Réclamation** : chaque détenteur d'index réflexes et créateur de SAFE peut réclamer une quantité fixe de chaque collatéral du système, calculée à partir du dernier prix de rédemption enregistré pour l'index concerné

5. Gouvernance

La grande majorité des paramètres seront immuables et les mécanismes internes des smart contracts ne pourront pas être modifiés, sauf si les détenteurs des jetons de gouvernance déploient un système entièrement nouveau. Nous avons opté pour cette stratégie afin d'éliminer les jeux d'influence que les gens peuvent avoir sur un processus de gouvernance pour servir leurs intérêts personnels, sapant ainsi la confiance accordée au système. Nous pensons donc que le bon fonctionnement du protocole passe par la réduction du facteur humain (« effet bitcoin ») pour maximiser l'adoption et minimiser les risques pour les développeurs souhaitant utiliser RAI dans leurs propres projets.

Pour sécuriser les quelques paramètres modifiables, nous proposons un module de gouvernance restreinte destiné à limiter les possibilités de modifications du système. Nous présentons également la glaciation de la gouvernance, qui consiste à empêcher toute modification de certaines parties du système par la gouvernance, passées certaines dates butoir.

5.1. Fréquence limitée des modifications via la gouvernance

L'une des caractéristiques du module de gouvernance restreinte est l'imposition de délais entre les modifications de valeur d'un même paramètre. Par exemple, le changement d'adresses des oracles utilisés dans le calculateur de médianes d'oracles (Section 6.2) n'est possible qu'après que T secondes se soient écoulées depuis la dernière modification de l'oracle.

5.2. Limitation des valeurs modifiables par la gouvernance

L'autre caractéristique de la gouvernance restreinte consiste à limiter l'amplitude des modifications de valeurs de chaque paramètre du système, de façon absolue ou relativement à une période donnée. C'est notamment le cas des des paramètres des versions initiales du mécanisme de rétroaction du taux de rédemption (section 4.2).

5.3. La glaciation de la gouvernance

La "glaciation" ("Ice Age") est un smart contract immuable qui impose des dates butoir sur le changement de certains paramètres du système et sur la mise à jour du protocole. Il permet à la gouvernance d'intervenir pour corriger des bugs avant que le protocole ne soit verrouillé et ne rejette toute intervention extérieure. La glaciation s'assure qu'un changement est autorisé en vérifiant le nom du paramètre et l'adresse du contrat concerné dans un registre des dates butoir. Si la date est dépassée, le changement est rejeté.

La gouvernance pourrait retarder la glaciation un certain nombre de fois si des bugs sont détectés à l'approche de la date du verrouillage du protocole. A titre d'exemple, il pourrait être décidé de limiter à trois reports la mise en oeuvre de la glaciation, à chaque fois pendant un mois, afin de disposer du temps nécessaire à l'implémentation et aux tests de correctifs de bugs.

5.4. Domaines essentiels où la gouvernance est nécessaire

Nous envisageons quatre domaines où la gouvernance pourrait être nécessaire, surtout lors des premières versions du protocole :

- **Ajout de nouveaux collatéraux** : RAI reposera uniquement sur l'ETH, mais d'autres index seront adossés à plusieurs types de collatéraux et la gouvernance permettra de diversifier les risques au fil du temps
- **Modification des dépendances externes** : les oracles et DEX dont dépend le système peuvent être mis à jour. La gouvernance peut diriger le système vers de nouvelles dépendances pour continuer à fonctionner correctement
- **Affiner les configurateurs de taux** : les paramètres des premiers régulateurs de politique monétaire pourront être modifiés par la gouvernance, dans certaines limites
- **Migration entre les versions du système** : dans certains cas, la gouvernance peut déployer un nouveau système, lui donner la permission d'émettre les jetons du protocole et retirer cette permission à l'ancien système. Cette migration est effectuée à l'aide du module de migration restreinte décrit ci-dessous.

5.4.1. Module de migration restreinte

Le changement de versions du système obéit à un mécanisme simple :

- Un registre de migration garde trace des systèmes couverts par le même jeton de protocole. A travers ce registre, il est possible d'autoriser ou de bloquer l'émission de nouveau jeton de protocole dans une enchère de dette pour un système donné
- A chaque déploiement d'une nouvelle version du système par la gouvernance, l'adresse du contrat de l'enchère de dette est envoyée au registre. La gouvernance doit indiquer si l'émission de jetons de protocole pourra ultérieurement être arrêtée. Elle peut également décider à tout moment si un système a la permission perpétuelle d'émettre des jetons, et donc ne pourra pas être migré vers un autre système
- Il y a une période de latence entre la proposition de déploiement d'un nouveau système et le retrait des permissions de l'ancien
- Un contrat optionnel peut être configuré de façon à stopper automatiquement un ancien système après qu'on lui ait retiré la permission d'émettre des jetons

Le module de migration peut être combiné à la fonctionnalité de glaciation afin de donner automatiquement à des systèmes particuliers la permission perpétuelle d'émettre des jetons.

6. Arrêt automatique du système

Certaines situations sont détectées par le système et déclenchent automatiquement un règlement, sans qu'il soit nécessaire de détruire les jetons de protocole :

- **Retard important dans un flux de prix** : le système détecte que certains flux de prix d'index ou de collatéraux n'ont pas été mis à jour depuis longtemps
- **Migration du système** : contrat optionnel ayant la capacité d'arrêter le protocole après une période de latence, qui débute au moment où la gouvernance retire la possibilité pour le mécanisme d'enchère de dette d'émettre des jetons (voir Module de migration restreinte)
- **Déviation constante du prix de marché** : le système détecte que le prix de marché de l'index a dévié pendant une longue période de *x%* par rapport au prix de rédemption.

La gouvernance aura la possibilité de mettre à jour ces fonctions d'arrêt jusqu'au moment où la glaciation verrouillera certaines parties du système.

7. Oracles

Le système a besoin d'accéder aux flux de prix de trois principaux types d'actifs : l'index, le jeton de protocole et chaque collatéral retenu. Les flux de prix peuvent être fournis par des oracles mis en place par la gouvernance ou par des réseaux d'oracles existants.

7.1. Oracles mis en place par la gouvernance

Les détenteurs de jetons de gouvernance ou l'équipe qui a lancé le protocole peuvent s'accorder avec d'autres entités qui collectent des flux de prix off-chain et les soumettre en une seule transaction à un smart contract chargé de calculer le point médian de ces données.

Cette approche permet de mettre à jour de façon très flexible l'infrastructure d'oracles, au prix d'un risque de contrepartie.

7.2. Le calculateur de médianes d'oracles

La calculateur de médianes d'oracles est un smart contract qui récupère les prix depuis de multiples sources hors du contrôle de la gouvernance (par exemple, un pool Uniswap V2 formé avec un collatéral de l'index et un autre stable coin), puis qui calcule le point médian de ces données. Le calculateur fonctionne de la façon suivante :

- Le calculateur maintient une liste des réseaux d'oracles qu'il peut interroger afin de récupérer des prix de collatéraux. Le calculateur est financé avec une partie de la valeur accumulée par le système (voir Gestion des index excédentaires, Section 11). Chaque réseau d'oracles exigeant des jetons spécifiques comme moyen de

paiement, le calculateur garde trace du montant minimum et du type de jeton nécessaires à chaque requête

- Pour être en mesure de d'injecter un nouveau flux de prix dans le système, tous les oracles doivent être appelés préalablement. Le calculateur commence par échanger une partie des intérêts accumulés dans le système contre l'un des jetons acceptés par l'oracle. Une fois l'oracle appelé, le calculateur catégorise l'appel comme "valide" ou "invalide". Si un appel est invalide, l'oracle fautif n'est plus appelé jusqu'à ce que tous les autres l'aient été et que le calculateur ait vérifié s'il y a une majorité valide. Un appel d'oracle valide est un appel accepté qui retourne un prix enregistré on-chain dans les dernières *m* secondes. Le mode de récupération est différent selon le contexte :
    - Pour les oracles "pull", d'où l'information peut être collectée immédiatement, le calculateur paie les frais et récupère directement le prix
    - Pour les oracles "push", le calculateur paie les frais, appelle l'oracle et attend un période spécifique avant de l'appeler de nouveau et d'accéder au prix
- Chaque résultat d'oracle est enregistré dans un tableau. Après que chaque oracle sélectionné ait été appelé et si le tableau contient assez de données pour former une majorité (par exemple, le calculateur a reçu des données valides depuis 3 oracles sur 5), les résultats sont ordonnés et le calculateur détermine le point médian
- Qu'une majorité ait ou non été trouvée, le tableau des résultats est ensuite remis à zéro et une période de *p* secondes s'écoule avant le processus recommence

7.2.1. Solution de secours du réseau d'oracle

La gouvernance peut utiliser une solution de secours pour injecter des prix dans le système au cas où le calculateur ne parvient pas à obtenir une majorité d'oracles valides plusieurs fois de suite.

Cette option doit être mise en place lorsque le calculateur est déployé, il est impossible de l'ajouter après coup. De plus, un smart contract supplémentaire surveille si la solution de secours est utilisée et arrête automatiquement le protocole si ce recours exceptionnel se prolonge anormalement.

8. Coffres-forts (SAFE)

Afin de générer des index, n'importe qui peut déposer et utiliser des crypto-actifs comme collatéral à l'intérieur de coffres-forts numériques appelés "SAFE". Tant qu'un SAFE est ouvert, sa dette augmente selon le taux d'emprunt du collatéral déposé. Quand le créateur d'un SAFE rembourse sa dette, cela lui donne le droit de retirer tout ou une partie du collatéral déposé .

8.1. Cycle de vie d’un SAFE

Il y a quatre étapes principales nécessaires depuis la création d’index réflexes jusqu’au remboursement de la dette d’un SAFE :

- Déposer des collatéraux dans le SAFE

  L'utilisateur doit d'abord créer un nouveau SAFE et y déposer un collatéral

- Générer des index adossés au collatéral du SAFE

  L'utilisateur spécifie le nombre d'index qu'il souhaite générer. Le système crée un montant égal de dette qui commence à s'accumuler selon le taux d’emprunt du collatéral

- Rembourser la dette d’un SAFE

  Lorsque le créateur SAFE souhaite retirer son collatéral, il doit rembourser la dette initiale plus les intérêts courus

- Retirer le collatéral

  Une fois que l'utilisateur a remboursé une partie ou la totalité de sa dette, il est autorisé à retirer le collatéral correspondant

9. Liquidation d’un SAFE

Afin de garder le système solvable et de couvrir la valeur totale de l'encours de la dette, chaque SAFE peut être liquidé dans le cas où son ratio de collatéralisation tombe sous un certain seuil. N'importe qui peut déclencher une liquidation qui conduit le système à confisquer le collatéral du SAFE et le vendre aux enchères.

9.1. Assurance contre la liquidation

Dans une version du système, les créateurs de SAFE peuvent avoir la possibilité de choisir un *déclencheur* pour la liquidation de leur SAFE. Les déclencheurs sont des smart contracts qui ajoutent automatiquement du collatéral dans le SAFE pour éviter sa liquidation, par exemple en vendant des positions *“short”* ou en interagissant avec des protocoles d'assurance tels que Nexus Mutual [6].

Une autre méthode pour protéger les SAFE est l'ajout de deux seuils de collatéralisation : safe et risk. Les utilisateurs de SAFE peuvent générer des dettes jusqu'à ce qu'ils atteignent le seuil safe (qui est plus élevé que risk) et ils ne sont liquidés que lorsque la collatéralisation du SAFE passe en dessous du seuil risk.

9.1.1. Enchères de liquidation

Pour initier une enchère de liquidation, le système utilise une variable appelée *liquidationQuantity* afin de déterminer la quantité de dette à couvrir et le montant

correspondant de collatéral à vendre. Une pénalité de liquidation est appliquée pour tout SAFE subissant une enchère.

9.1.2. Paramètres des enchères de liquidation

Paramètre	Description
minimumBid	
discount	
lowerCollateralMedianDeviation	
upperCollateralMedianDeviation	
lowerSystemCoinMedianDeviation	
upperSystemCoinMedianDeviation	
minSystemCoinMedianDeviation	

9.1.3. Mécanisme d'enchère de liquidation

L'enchère à réduction fixe est un moyen simple (par rapport aux enchères anglaises) de régler la dette non remboursée en mettant en vente le collatéral en échange de jetons du système. Les acheteurs participant à l'enchère doivent autoriser le transfert de leur safeEngine.coinBalance pour ensuite appeler buyCollateral qui permet d'échanger des jetons du système contre un collatéral vendu à prix réduit par rapport au dernier prix de marché enregistré.

Les acheteurs participant à l'enchère peuvent également accéder aux collatéraux associés à une enchère spécifique en faisant appel aux fonctions getCollateralBought ou getApproximateCollateralBought. Notons que le getCollateralBough n'est pas une vue statique mais provient du redemptionPrice des oracles, alors que getApproximateCollateralBought utilise le dernier prix de rédemption (lastReadRedemptionPrice).

9.2. Enchère de dettes

Dans le cas où une enchère de liquidation échoue à couvrir la dette d'un SAFE, et si le système ne dispose pas d'excédents en réserve, n'importe qui peut déclencher une enchère de dette. Ces enchères consistent à émettre des jetons de protocoles (voir Section "Jetons de Protocoles") et les vendre pour des index afin d'annuler la créance irrécouvrable du système.

Pour lancer une enchère de dette, le système doit utiliser deux paramètres:

- initialDebtAuctionAmount : la quantité initiale de jetons de protocole à créer après la vente aux enchères

- detteAuctionBidSize : la taille de l'enchère initiale (combien d'index doivent être proposés en échange de *initialDebtAuctionAmount* jetons de protocole)

9.2.1. Paramétrage des enchères de dette autonomes

Le montant initial de jetons du protocole à créer lors d'une enchère de dette peut soit être défini par un vote de gouvernance, soit être ajusté automatiquement par le système. Une version automatisée devrait être intégrée aux oracles (Section 6) à partir desquels le système lirait les prix de marché des jetons du protocole et ceux des index réflexes. Le système définirait alors la quantité initiale de jetons de protocole (*initialDebtAuctionAmount)* à créer pour la quantité *debtAuctionBidSize* d'index. *initialDebtAuctionAmount* peut être fixé avec une décote en comparaison du prix de marché réel jeton de protocole / index, afin d'inciter les acheteurs à participer à l'enchère.

9.2.2. Paramètres des enchères de dette

Paramètre	Description
amountSoldIncrease	Surplus de jetons de protocoles à émettre pour une même quantité d'index
bidDecrease	Sous-enchère minimum
bidDuration	Combien de temps l'offre dure après avoit été soumise (en secondes)
totalAuctionLength	Durée totale de l'enchère (en secondes)
auctionsStarted	Nombre d'enchères démarrées jusqu'à présent

9.2.3. Mécanisme d'enchère de dette

Contrairement aux enchères de collatéraux, les enchères de dette n'ont qu'une seule étape :

decreaseSoldAmount(uint id, uint amountToBuy, uint bid) : diminuer la quantité de jetons de protocole acceptés en échange d'une quantité donnée d'index.

L'enchère est relancée si aucune offre n'est faite. A chaque fois qu'elle est relancée, le système augmente le nombre de jetons de protocole offerts pour une même quantité d'index. La nouvelle quantité de jetons est calculée selon la formule *lastT okenAmount * amountSoldIncrease / 100*. Une fois l'enchère conclue, le système émet les jetons et les attribue au meilleur offreur.

10. Jetons de protocole

Comme décrit précédemment, chaque protocole doit être sécurisé par un jeton émis via des enchères de dette. En dehors de cette fonction de sécurisation, le jeton sera utilisé pour piloter certains composants du système. La quantité de jetons de protocole sera

graduellement réduite via les enchères sur les excédents. La quantité de réserves excédentaires qui doit être accumulée dans le système (à partir des intérêts perçus) avant que les enchères ne soient déclenchées est appelée le buffer d'excédent (*surplusBuffer*) et est automatiquement ajustée en tant que pourcentage de la dette totale.

10.1. Fonds d'assurance

En dehors du jeton de protocole, la gouvernance peut créer un fonds d'assurance détenant un large portefeuille d'actifs non collatéraux pouvant servir de filet de sécurité pour les enchères de dettes.

10.2. Enchères d'excédents

Les enchères d'excédents consistent à céder les intérêts accumulés dans le système en échange de la destruction de jetons de protocole excédentaire.

10.2.1. Paramètres des enchères d'excédents

Paramètre	Description
bidIncrease	Surenchère minimum
bidDuration	Combien de temps l'offre dure après avoit été soumise (en secondes)
totalAuctionLength	Durée totale de l'enchère (en secondes)
auctionsStarted	Nombre d'enchères démarrées jusqu'à présent

10.2.2. Mécanisme des enchères d'excédents

Les enchères d'excédent se déroulent en une seule phase :

`increaseBidSize(uint id, uint amountToBuy, uint bid)` : toute personne peut surenchérir sur la quantité de jetons de protocole offerts pour un même nombre d'index (l'excédent). Chaque nouvelle offre doit être égale ou supérieure à *lastBid * bidIncrease / 100*. L'enchère se termine après que *totalAuctionLength* secondes se soient écoulées, ou bien lorsqu'aucune nouvelle offre n'ait été proposée dans un délai de *bidDuration* secondes après la dernière offre.

Une enchère redémarre si aucune offre n'a été faite. Lorsqu'une enchère se termine avec au moins une offre, l'excédent est attribué à la meilleure offre et les jetons de protocoles offerts sont détruits.

11. Gestion des index excédentaires

A chaque fois qu'un utilisateur génère des index et crée implicitement de la dette, le système applique un taux d'emprunt à son SAFE. L'intérêt accumulé est distribué dans deux smart contracts différents :

- Le *mécanisme de comptabilisation* utilisé pour déclencher des enchères de dette (Section 9.2) et d'excédents (Section 10.1)
- La *trésorerie excédentaire* utilisée pour financer des composants de l'infrastructure comme le module d'Oracle et encourager des acteurs externes à participer au système

La trésorerie excédentaire finance le module d'Oracle (Section 6). Selon le cas, la trésorerie paie les oracles off-chain sélectionnés par la gouvernance, ou bien paie les appels vers des réseaux d'oracles. Elle peut également être utilisée pour rembourser les adresses qui paient le gas lors d'appels d'oracles.

La trésorerie est également utilisée pour encourager des acteurs externes à participer au fonctionnement du système, par exemple en sélectionnant de nouveaux types de collatéraux ou en réglant les taux du marché monétaire (Section 4.2)

La trésorerie peut être configurée de façon à ce que certains acteurs externes soient écartés du financement et que d'autres prennent leur place.

12. Acteurs externes

Le système dépend d'acteurs externes pour fonctionner correctement. Ces acteurs sont encouragés économiquement à participer aux enchères, au traitement du règlement global, au market making et à la mise à jour des flux de prix afin de maintenir un fonctionnement optimal du système.

Nous fournirons les interfaces utilisateurs initiales et les scripts d'automatisation afin de permettre la participation la plus large possible au bon fonctionnement et à la sécurité du système.

13. Marchés ciblés

Nous pensons que RAI sera utile dans deux domaines principaux :

- Diversification de portefeuille : les investisseurs pourront obtenir une exposition atténuée à des crypto-actifs volatiles comme l'ETH

- Collatéral d'actifs synthétiques : RAI permet à des protocoles comme UMA, MakerDAO ou Synthetix une exposition moins risquée aux marchés crypto et donne aux utilisateurs plus de temps pour sortir de leurs position en cas de chute brutale des marchés déclenchant des liquidations massives (comme dans le cas du Jeudi Noir de mars 2020)

14. Domaines de recherche futurs

Afin de repousser les limites de la monnaie décentralisée et d'innover en matière de finance décentralisée, nous continuerons à étudier des alternatives dans des domaines clés comme la minimisation de la gouvernance et les mécanismes de liquidation.

Nous voulons d'abord poser les fondations de standards futurs pour les protocoles protégés de toute tentative de prise de contrôle externe, et pour de véritables "robots monétaires" capables de s'adapter de façon autonome en réaction aux forces du marché. Nous inviterons ensuite la communauté Ethereum à débattre et à concevoir des améliorations à nos propositions, en particulier en ce qui concerne les enchères de dettes et de liquidation.

15. Gestion des risques

Le développement et le lancement d'un index réflexe s'accompagnent de nombreux risques :

- **Bugs dans les smart contracts** : le risque majeur pour le système est la possibilité d'un bug qui permettrait à n'importe qui de retirer tout le collatéral ou qui verrouillerait définitivement le protocole. Nous prévoyons de faire auditer notre code par plusieurs spécialistes en sécurité et de le mettre à disposition sur un testnet avant tout déploiement en production

- **Echec d'un oracle** : nous agrégerons les flux de prix de multiples réseaux d'oracles et nous mettrons en place des règles strictes pour limiter leur mise à jour afin d'empêcher toute possibilité d'introduction malicieuse de prix erronés par la gouvernance

- **Evénements de type "cygne noir"** : il existe un risque de chute brutale du prix du collatéral, qui peut à son tour déclencher des liquidations massives de SAFE. Ces liquidations pourraient s'avérer insuffisantes pour couvrir la dette. Le système modifierait alors son buffer d'excédent (*surplusBuffer*) afin d'assurer une couverture minimale de la dette et d'absorber les chocs du marché

- **Valeurs incorrectes pour les taux du marché monétaire** : les mécanismes de rétroaction autonomes sont expérimentaux et pourraient ne pas se comporter comme les simulations le prédisent. Nous prévoyons d'autoriser la gouvernance à modifier dans certaines limites ces paramètres afin de pouvoir réagir à des scénarios inattendus

- **Défaillance du marché de liquidation** : des liquidateurs actifs et en nombre suffisant sont indispensables à la couverture de la dette du système. Nous prévoyons de créer des interfaces utilisateurs et des scripts automatisés pour faciliter l'accès au plus grand nombre de liquidateurs

16. Synthèse

Nous proposons un protocole qui s'autonomise progressivement vis-à-vis de tout contrôle manuel et qui émet un actif collatéralisé et peu volatile appelé index réflexe. Nous avons d'abord présenté le mécanisme autonome conçu pour influencer le prix de marché de l'index, puis décrit comment une combinaison de smart contracts limite le pouvoir que les détenteurs de jetons ont sur le système. Nous avons tracé les contours d'un mécanisme autonome pour calculer le prix médian à partir de flux de prix provenant de multiples réseaux d'oracles, et terminé en présentant le mécanisme général d'émission des index et de liquidation des collatéraux.

17. Références

[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] K.J. Åström, R.M. Murray, "Feedback Systems: An Introduction for Scientists and Engineers", https://bit.ly/3bHwnMC

[5] R.J. Hawkins, J.K. Speakes, D.E. 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

18. Glossaire

Index réflexe : un actif collatéralisé atténuant la volatilité de son sous-jacent

RAI : notre premier index réflexe

Prix de rédemption : le prix cible de l'index. Il change en fonction du taux de rédemption (calculé par le MRTR), lorsque le prix de marché en est éloigné. A pour but d'inciter les créateurs de SAFE à générer plus d'index ou à rembourser une partie de leur dette

Taux d'emprunt : taux d'intérêt annuel appliqué à tous les SAFE ayant une dette en cours

Mécanisme de rétroaction sur le taux de rédemption (MRTR) : un mécanisme autonome qui compare le prix de marché au prix de rédemption d'un index réflexe et calcule un taux de rédemption incitant les créateurs de SAFE à générer plus ou moins de dette (et tente implicitement de minimiser la déviation entre prix de marché et prix de rédemption)

Configurateur de marché monétaire : un mécanisme similaire au MRTR qui manipule plusieurs paramètres monétaires à la fois. Dans le cas des index réflexes, il modifie le taux d'emprunt et le prix de rédemption

Calculateur de médianes d'oracles (CMO) : un smart contract qui collecte des prix depuis plusieurs réseaux d'oracles non contrôlés par la gouvernance, et qui calcule leur valeur médiane si une majorité d'oracles produisent un résultat valide

Module de gouvernance restreinte : un ensemble de smart contracts limitant le pouvoir que les détenteurs de jetons de gouvernance ont sur le système. Il permet de limiter la fréquence de modification manuelle de certains paramètres et de borner les valeurs que la gouvernance peut leur affecter

Glaciation de la gouvernance : smart contract non modifiable ayant pour but de déclencher le verrouillage de la plupart des composants du protocole au-delà d'une date butoir, afin d'empêcher définitivement toute intervention extérieur sur son fonctionnement

Mécanisme de comptabilisation : composant en charge du déclenchement des enchères de dettes et d'excédents. Il suit la quantité de dette mise en enchère, de la dette irrécouvrable ne trouvant pas preneur via une enchère et le buffer d'excédent (*surplusBuffer)*

Buffer d'excédent (*surplusBuffer*) : part de l'intérêt captée et accumulée dans le système. Tout intérêt généré au-dessus de cette limite est cédé via des enchères d'excédents contre des jetons de protocoles qui sont ensuite détruits

Trésorerie d'excédent : smart contract permettant à différents modules du système de retirer des intérêts accumulés (par exemple le CMO, pour payer les appels aux oracles)