لوبرنج:
بروتوكول تداول العملة الرمزية اللامركزيه

دانيال وانج
daniel@loopring.org

جي زهو
jay@loopring.org

أليكس وانج
alex@loopring.org

ماثيو فينيستون
matt.finestone@gmail.com

https://loopring.org

٢٥ نيسان ٢٠١٨

ملخص

لوبرنج هو بروتوكول مفتوح لبناء منصات التداول اللامركزية. لوبرنج يعمل كمجموعة عامة من العقود الذكية المسؤولة عن التداول والتسوية ، مع مجموعة غيرمرتبطة بالسلسلة من الجهات الفاعلة التي تقوم بتجميع الأوامر وتوصيلها. البروتوكول مجاني وقابل للتوسعة ويعمل كأساس بناء الكتلة للتطبيقات اللامركزية (dApps) التي تتضمن وظائف منصات التداول. تعمل معاييره القابلة للتشغيل البيني على تسهيل التداول المجهول. ومن التحسينات المهمة على بروتوكولات منصه التداول اللامركزية الحالية القدرة على مزج الطلبات ومطابقتها مع الأوامر الأخرى الغير متشابهة ، وتجنب قيود أزواج التداول ثنائي العملة الرمزية وتحسن السيولة بشكل كبير. توظف لوبرنج أيضاًا حلاًا فريداً وقوياًا لمنع تشغيل الواجهة: وهي محاولة غير عادلة لإرسال المعاملات إلى كتلة أسرع من مزود الحلل الأصلي. لوبرنج قائم على البلوكشين ، وقابل للانتشار على أي بلوكشين مع وظيفة العقد الذكي. في وقت هذه الكتابة ، انه قابل للتشغيل على اثريوم [22] [5] و [7] Qtum مع [3] NEO قيد الإنشاء.

1. المقدمة

مع انتشار الأصول القائمة على البلوكشين ، الحاجة إلى تداول هذه الأصول بين الأطراف المختلفه ازداد بشكل ملحوظ. مع طرح آلاف العملات الرمزية الجديدة - بما في ذلك ترميز الأصول التقليدية - هذه الحاجة اصبحت كبيرة. سواء كان تبادل العملات الرمزية لدوافع تداول المضاربة ، أو التحويل إلى شبكات الوصول عبر فائدة العملات الرمزية الأصلية الخاصة بها ، فإن القدرة على تداول واحد من الاصول المشفرة لآخرى هو الأساس للنظام الأكبر. في الواقع ، هناك طاقة محتملة للأصول [9] ، وتحقيق هذه الطاقة - فتح رأس المال - يتطلب ليس فقط تأكيد الملكية ، التي تسمح تقنيات البلوكشين بثباتة ، ولكن القدرة على نقل هذه الأصول وتحويلها بحرية. على هذا النحو ، فإن تداول العملات الرمزية (القيمة) هي حالة استخدام مقنعة لتكنولوجيا البلوكشين. حتى الآن ، ومع ذلك ، اتفق عشاق التشفير إلى حد كبير بتداول العملات الرمزية على منصات التداول المركزية التقليدية. هناك حاجة إلى بروتوكول لوبرنج لأنه ، كما البيتكوين [13] يؤكد بشكل طوعي على ، ما يتعلق بالنقد الإلكتروني للاقران (peer-to-peer) ، تضيع الفوائد الرئيسية إذا كان لا يزال هناك حاجةٌ إلى طرف ثالث موثوق به لمنع الإنفاق المزدوج ، لذلك أيضا الفوائد الرئيسية للأصول اللامركزية تفقد إذا كان يجب أن تمر عبر منصات تداول مركزية موثوقة ومبنية. إن تداول العملات الرمزية اللامركزية في منصات التداول المركزية لا معنى له من الناحية الفلسفية ، حيث تفشل في دعم الفضائل التي تتبناها هذه المشاريع اللامركزية. هناك أيضاًا العديد من المخاطر والقيود العملية في استخدام منصات التداول [11] [1] [15] وفي كثير من الحالات نجحت في التخفيف من المخاطر الأمنية باستخدام تقنيات البلوكشين لالغاء الوساطة. ومع ذلك ، عندما تصبح قابلية DEX كبنية أساسية حاسمة للاقتصاد الجديد ، هناك مجال كبير لتحسين الأداء. يهدف Loopring إلى توفير أدوات معيارية للبنية التحتية المذكورة من خلال dApp بروتوكولها المفتوح القائم على تطبيقات dApp.

2. مشهد التداول الحالي

١.٢ أوجة القصور في منصات التداول المركزية

المخاطر الرئيسية الثلاثة للمنصات التداول المركزية هي ؛ ١) انعدام الأمن ، ٢) انعدام الشفافية ، و ٣) نقص السيولة. ينشأ انعدام الأمن عن المستخدمين الذين يسلمون عادة السيطرة على مفاتيحهم الخاصة (الأموال) إلى كيان مركزي واحد. هذا يعرض المستخدمين لاحتمال أن منصات التداول المركزية تقع فريسة للقراصنة الاشرار. إن مخاطر الأمن والقرصنة التي تواجه جميع منصات التداول المركزية معروفة جيداً [12] [10] ، ومع ذلك يتم قبولها في كثير من الأحيان على أنها حصص مخططة لتداول العملات الرمزية. لا تزال منصات التداول المركزية تمثل مصائد مخترقة لهجمات القراصنة لأن خوادمهم تحتفظ بملايين الدولارات من أموال المستخدمين. يمكن لمطوري منصات التداول أيضاً إجراء أخطاء بريئة وعرضية مع أموال المستخدمين. ببساطة ، لا يتحكم المستخدمون في العملات الرمزية الخاصة بهم عند إيداعها في منصه تداول مركزية. ويؤدي الافتقار إلى الشفافية إلى تعرض المستخدمين لخطر حدوث ان منصات تداول معروفه تتصرف بشكل غير عادل. ويكمن الفرق هنا في عوامل سوء تشغيل منصة التداول ، حيث لا يقوم المستخدمون فعلياً بالتداول في أصولهم الخاصة في منصات التداول المركزية ، ولكن بالأحرى يستخدمون IOU. عندما يتم إرسال العمله الرمزية إلى محفظة منصة التداول ، فإن منصة التداول تأخذ الوصاية ، وتقدم IOU. جميع منصات التداول بعد ذلك تتم بشكل فعال بين ال IOU للمستخدمين. من أجل السحب ، يسترد المستخدمون قيمة نظام IOU الخاص بهم من خلال منصات التداول ، ويستلمون العملات الرمزية إلى عنوان المحفظة الخارجية الخاصة بهم. خلال هذه العملية ، يوجد نقص في الشفافية ، ويمكن أن يتم إغلاق الحساب ، أو تجميد حسابك ، أو الإفلاس ، إلخ. ومن الممكن أيضاً أن يستخدموا أصول المستخدمين لأغراض أخرى أثناء الاحتجاز ، مثل إقراضهم لأطراف ثالثة. يمكن أن يؤدي الافتقار إلى الشفافية إلى تكبد المستخدمين دون فقد كامل للأموال ، كما هو الحال في رسوم منصات التداول المرتفعة ، والتأخير في ذروة الطلب ، والمخاطر التنظيمية ، والأوامر التي يتم تنفيذها على المستوى الأمامي. نقص السيولة. من وجهة نظر مشغلي منصات التداول ، السيولة المجزأة تمنع تسجيل الدخول في منصات التداول الجديدة بسبب سيناريوهين يستلزمان الفوز. أولاً ، تفوز منصات التداول مع أكبر عدد من أزواج التداول ، لأن المستخدمين يجدون أنه من المرغوب فيه إجراء جميع صفقاتهم في منصة تداول واحدة. ثانياً ، تفوز منصات التداول مع أكبر دفتر طلبيات ، بسبب فروق الأسعار المرغوبة عند كل زوج تداول. هذا لا يشجع المنافسة من الوافدين الجدد لأنه من الصعب عليهم بناء السيولة الأولية. ونتيجة لذلك ، فإن العديد من منصات التداول تسيطر على حصة كبيرة من السوق على الرغم من شكاوى المستخدمين وحتى حوادث الاختراق الرئيسية. تجدر الإشارة إلى أنه مع فوز منصات التداول المركزية بحصتها في السوق ، فإنها تصبح هدف اختراق دائماً و في أي وقت. من وجهة نظر المستخدمين ، فإن السيولة المجزأة تقلل إلى حد كبير من تجربة المستخدم. في منصات التداول المركزي ، يمكن للمستخدمين التداول فقط داخل صناديق السيولة الخاصة بمنصات التداول ، ضد دفتر الطلبات الخاص بهم ، وبين أزواج العملات الرمزية المدعومة. للتداول في العملة الرمزية A من أجل العملة الرمزية B ، يجب على المستخدمين الذهاب إلى منصة تداول تدعم كلا العملات الرمزية أو التسجيل في منصات التداول المختلفة ، والكشف عن المعلومات الشخصية. غالباً ما يحتاج المستخدمون إلى تنفيذ الصفقات الأولية أو المتوسطة ، عادةً مقابل ال BTC أو الETH، ودفع فروق أسعار العطاء في العملية. وأخيراً ، قد لا تكون سجلات الطلبات عميقة بما فيه الكفاية لإتمام الصفقة دون أي انزلاق مادي. حتى إذا كانت منصة التداول تهدف إلى معالجة كميات كبيرة ، فليس هناك ما يضمن

أن هذا الحجم والسيولة ليسا مزيفين [14]. والنتيجة هي مستودعات سيولة منفصلة ونظام مجزأ يشبه النظام المالي القديم ، مع حجم تداول مركزي هام على عدد قليل من منصات التداول . إن وعود السيولة العالمية بالحصانات لا تحمل أي ميزة داخل منصات التداول المركزية.

٢.٢ عدم ملاءمة منصات التداول اللامركزية

تختلف منصات التداول اللامركزية عن منصات التداول المركزية جزئياً لأن المستخدمين يحتفظون بالسيطرة على مفاتيحهم الخاصة (الأصول) عن طريق تنفيذ الصفقات مباشرة على البلوكشين الأساسي. من خلال الاستفادة من التكنولوجيا الامنة للعمل المشفرة نفسها ، فإنها تخفف بنجاح العديد من المخاطر المذكورة أعلاه المحيطة بالأمن. ومع ذلك ، فإن المشاكل لا تزال قائمة فيما يتعلق بالأداء والقيود الهيكلية .

غالبا ما تظل السيولة مشكلة حيث يجب عل المستخدمين البحث عن الأطراف المقابلة عبر مجموعات و معايير السيولة المتفاوتة. تتوجد تأثيرات السيولة المجزأة إذا لم تستخدم DEXs أو dApps بشكل عام معايير متسقة للتشغيل المتداخلالمتداخل ، وإذا لم يتم مشاركة / نشر الطلبات عبر شبكة واسعة. يمكن أن تؤثر سيولة سجلات الأوامر المحدودة ، وعلى وجه التحديد ، على المرونة - مدى سرعة تجديد حد الاوامر المنفذة - بشكل ملحوظ على استراتيجيات التداول الأمثل [2]. إن غياب مثل هذه المعايير لم يؤد فقط إلى انخفاض السيولة ، بل إلى التعرض لمجموعة من العقود الذكية غير المحمية التي قد تكون غير آمنة.

علاوة على ذلك ، بما أن عمليات التداول تتم على سلسلة ، فإن DEXs تمثل حدود البلوكشين الأساسي ، وهي: قابلية التوسع ، والتأخير في التنفيذ (التعدين) ، والتعديلات المكلفة على الطلبات. وبالتالي ، فإن سجلات أوامر البلوكشين لا تتوسع بشكل جيد ، حيث أن تنفيذ الشفرة على البلوكشين يتكبد تكلفة (جاز) ، مما يجعل من الإلغاء المتعدد لأوامر أمرًا باهظًا. وأخيرا، لأن سجلات اوامر البلوكشين علنية، المعاملة لتضع امرا ما فانه يكون مرئيا من قبل المنقبين حيث انها تنتظر حتى تنقب الى الكتلة التاليه لها وتوضع في سجل الاوامر. هذا التأخير يعرض المستخدم لخطر أن يتجة للامام وأن يتحرك السعر أو التنفيذ ضده.

٣.٢ الحلول الهجينة

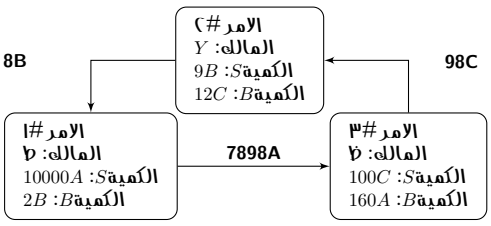
للأسباب المذكورة أعلاه ، فإن منصات التداول القائمه على البلوكشين البحت لديها قيود تجعلها غير قادر على المنافسة مع منصات التداول ات المركزية. هناك مقايضة بين الثقة الممثلة في السلسلة ، وسرعة منصات التداول المركزي ومرونة الطلب. تمد البروتوكولات مثلLoopring و 0x [21] حلاً للتسوية على السلسلة بإدارة أوامر خارج السلسلة. تدور هذه الحلول حول العقود الذكية المفتوحة ، ولكنها تتخطى حدود قابلية التوسع من خلال تنفيذ عدة وظائف خارج السلسلة وإعطاء نقاط مرونة في تحقيق الأدوار المهمة للشبكة. نهجنا لحل هجين من خلال هذه الورقة. مع ذلك ، لا تزال هناك عيوب للنموذج الهجين أيضاً [4]. يقترح بروتوكول Loopring اختلافات ذات مغزى في نهجنا لحل هجين من خلال هذه الورقة.

3. بروتوكول *Loopring*

Loopring ليس DEX، بل هو عبارة عن بروتوكول نموذجي لبناء DEXs على العديد من تقنيات البلوكشين. نقوم بتفكيك الأجزاء المكونة لمنصة التداول التقليديه ونقدم مجموعة من العقود الذكية العامة والجهات الفاعلة اللامركزية في مكانها. وتشمل الأدوار في الشبكة المحافظ ، والمرحلات ، بلوكشين الكونسورتيوم في تقاسم السيولة ، ومستعرضين سجلات الطلبات ، وحلقة المنقبين ، وخدمات ترميز الأصول. قبل تحديد كل منهما ، يجب علينا أولاً أن نفهم أوامر Loopring.

١.٣ ترتيب الطوق

يتم التعبير عن أوامر Loopring في ما نسميه نموذج أمر أحادي الاتجاه UDOM .(UDOM) [20] يعبر عن الطلبات كطلبات تبادل العملة الرمزية ، المبلغ بيع \ المبلغ شراء ، (المبلغ مقابل البيع \ الشراء) بدلاً من المزايدات ويسأل. نظرًا لأن كل طلب هو سعر صرف بين عملتين رمزيتين ، فإن الميزة القوية للبروتوكول هي مزج ومطابقة الأوامر المتعددة في دائرة التداول. باستخدام ما يصل إلى ٦١ طلباً بدلاً من زوج تداول واحد ، هناك زيادة كبيرة في السيولة وإمكانية تحسين الأسعار.



١.٣ شكل: حلقة الطلب لـ٣ أوامر

يوضح الشكل أعلاه حلقة الطلب من ٣ أوامر. لكل طلب لبيع العملة الرمزية (العملات الرمزية بيع) هو طلب لشراء العملة الرمزية الآخرى (العملة الرمزية شراء). يقوم بإنشاء حلقة تسمح لكل أمر بتبادل العملات الرمزية المطلوبة دون الحاجة إلى طلب متعارض لزوجها. وبالطبع ، لا يزال من الممكن تنفيذ اوامر تداول الأزواج التقليدية ، في ما يتعلق بشكل أساسي بحلقة الطلب.

تعريف ١.٣. (حلقة الطلب)

دع $C_0, C_1, ..., C_{n-1}$ لتكن عدد n من العملات الرمزية المختلفة ،

$O_{0\to1}, ..., O_{i\to i\oplus1}, ..., O_{n-1\to0}$ لتكن عدد n من اوامر التداول. يمكن لهذه الاوامر تشكيل حلقة طلب للتداول:

$$O_{0\to1} \to ... \to O_{i\to i\oplus1} \to ... \to O_{n-1\to0}$$

حيث n هو طول حلقة الطلب، و $i \oplus 1 \equiv i+1 \mod n$

تكون حلقة الطلب صالحة عندما يمكن تنفيذ جميع المعاملات المركبة بسعر صرف يساوي أو أفضل من المعدل الأصلي المحدد ضمنيًا من قبل المستخدم. للتحقق من صلاحية حلقة الطلب ، يجب أن تتلقى العقود الذكية لبروتوكول لوبرنج حلقات الاوامر من منقبين الحلقة حيث يكون سعر أسعار الصرف الأصلية لجميع الطلبات مساوياً أو أكبر من ١.

لنفترض أن أليس و بوب يرغبان في تبادل العملة الرمزية A و B. أليس لديها 15 عملة رمزية من A وتريد 4 عمل رمزية من B لها ؛ يملك بوب 10 عمل رمزية B ويريد 30 عملة رمزية A له.

من يشتري ومن يبيع؟ هذا يعتمد فقط على الأصل الذي نقوم بتحديدة لتقديم عرض للأسعار. إذا كانت العملة الرمزية A هي المرجع ، فإن اليس ستشتري العملة الرمزية B بسعر $A3.75 = \frac{15}{4}$ ، بينما سيبيع بوب العملة الرمزية B بسعر $A3.00 = \frac{30}{10}$. في حالة تحديد العملة الرمزية B كمرجع ، نفرض أن أليس ستبيع 15 عملة رمزية A بسعر $B0.26666667 = \frac{4}{15}$ و بوب سيشتري 10 عملة رمزية A بسعر $B0.33333334 = \frac{10}{30}$. وبالتالي ، من هو المشتري أو البائع اعتباطيا.

في الحالة الأولى ، تكون اليس على استعداد لدفع سعر أعلى $(3.75A)$ من السعر الذي يبيعه بوب لعملاته الرمزية مقابل $(3.00A)$ ، بينما في الحالة الثانية ، يكون بوب على استعداد لدفع سعر أعلى $(0.33333334B)$ من السعرالذي تقوم أليس ببيعه لعملاتها الرمزيه مقابل $(0.26666667B)$. من الواضح أن التداول ممكن عندما يرغب المشتري في دفع سعر مساوي أو أعلى من سعر البائع.

$$\frac{\frac{15}{4}}{\frac{30}{10}} = \frac{\frac{10}{30}}{\frac{4}{15}} = \frac{15}{4} \cdot \frac{10}{30} = 1.25 > 1 \qquad (١)$$

وبالتالي ، للمقدرة على تنفيذ مجموعة من الأوامر n ، بشكل كامل أو جزئي ، نحتاج إلى معرفة ما إذا كان ناتج كل واحد من أسعار

الصرف مثل نتائج أوامر الشراء برقم أكبرمن أو يساوي ١. إذا كان الأمر كذلك ، يمكن أن تكون الأوامر n منفذه جزئياًا أو كلياًا [16].

إذا قمنا بإدخال نظير ثالث ، تشارلي ، بحيث ترغب أليس في إعطاء X_1 للعملة الرمزية A وتحصل على Y_1 من العملة الرمزية B ، بوب يريد إعطاء X_2 للعملة الرمزية B ويحصل على Y_2 للعملة الرمزية C ، وتريد تشارلي اعطاء X_3 للعملة الرمزية C وتحصل على Y_3 للعملة الرمزية A . العملات الرمزية الضرورية موجودة ، والتداول ممكن إذا:

$$\frac{x_1 \cdot x_2 \cdot x_3}{y_1 \cdot y_2 \cdot y_3} \geq 1 \qquad (٢)$$

انظر القسم 7.1 لمزيد من التفاصيل حول أوامر لوبرنج

4. المشتركين في النظام

يزود المشتركون في النظام بشكل مشترك جميع الوظائف التي تقدمها منصات التداول المركزي.

- **المحافظ** : خدمة أو واجهة المحفظة العامة التي تمنح المستخدمين إمكانية الوصول إلى العملات الرمزية الخاصة بهم والطريق لإرسال اوامر إلى شبكة Loopring . سيتم تحفيز المحافظ لإنتاج الطلبات من خلال تقاسم الرسوم مع منقبين الحلقة (انظر القسم ٨). مع الإعتقاد بأن مستقبل التداول سيحدث مع أمان محافظ المستخدم الفردي ، فإن ربط صناديق السيولة هذه من خلال بروتوكولنا هو أمر بالغ الأهمية.

- **تحالف مشاركة سيولة البلوكشين\ شبكة الترحيل(التبديل)**: شبكة التبادل لتقاسم الامر والسيولة. عندما يقوم العقد بتشغيل برنامج تبادل لوبرنج ، فإنها تكون قادرة على الانضمام إلى الشبكة الموجودة وتشارك السيولة مع المرحلات الأخرى عبر البلوكشين المجمع. تحالف البلوكشين الذي نقوم ببناءه كأول تنفيذ لديه مشاركة طلب في الوقت الفعلي تقريبا (الكتل من ثانية إلى ثانيتن) ، وتزيل السجل القديم للسماح بتنزيل أسرع من خلال العقد الجديدة. بشكل خاص ، لا تحتاج المبدلات إلى الانضمام إلى هذا الاتحاد. يمكنهم العملات بمفردهم وعدم تقاسم السيولة مع الآخرين ، أو يمكنهم بدء وإدارة شبكة تقاسم السيولة الخاصة بهم.

- **المبدلات \منقبين الحلقة** (Relays/Ring-Miners) : المبدلات او المرحلات هي العقد التي تتلقى الاوامر من المحافظ أو شبكة المرحلات ، تحافظ على سجل حجوزات الطلب العامة والتداول ، و بث الأوامر اختياريا إلى المرحلات الأخرى (عن طريق أي وسط عشوائي خارج السلسلة) و \ أو عقد شبكة الترحيل. تعدين الحلقة هي ميزة - وليس شرطا - للمرحلات. وهو ثقيل حسابيا ويتم خارج السلسلة تماماً. نسمي المرحلات بخاصية تعدين الحلقة التي يتم تشغيلها على منقبين الحلقة ، الذين ينتجون حلقات الاوامر عن طريق تجميع الأوامر المتباينة معا . تكون المرحلات مجانية في (١) كيفية اختيارها للتواصل مع بعضها البعض ، (٢) كيفية بناء سجلات اوامرها ، و (٣) كيفية تعدين حلقات الاوامر (خوارزميات التعدين).

- **العقود الذكية لبروتوكول لوبرنج** (LPSC) : مجموعة من العقود الذكية العامة و المجانية التي تتحقق من حلقات الاوامر المتلقاة من المنقبين ، تسوي وتحول العملات الرمزية نيابة عن المستخدمين بطريقة امنة ، وتكافيء

منقبين الحلقة والمحافظ برسوم ، وتصدر الأحداث. مستعرضي المرحلات \الطلبات يستمعوا إلى هذه الأحداث لابقاء سجلات حجز الطلبات و التداول. انظر الملحق (أ) للحصول على التفاصيل.

- **خدمات ترميز الأصول** (ATS) : الجسر بين الأصول التي لا يمكن تداولها مباشرة على لوبرنج . انها خدمات مركزية تدار من قبل شركات أو منظمات جديرة بالثقة. يقوم المستخدمون بإيداع الأصول (مادية أو عملات ورقية أو عمل رمزية من سلاسل أخرى) ويحصلوا على العملات الرمزية التي تم إصدارها ، والتي يمكن استبدالها بالإيداع في المستقبل. لا يعد لوبرنج بروتوكول تداول عبر السلسلة (حتى يوجد حل مناسب) ، ولكن ATS تمكن من تداول العملات الرمزية ERC20 [18] مع الأصول المادية وكذلك الأصول على تقنيات البلوكشين الأخرى.

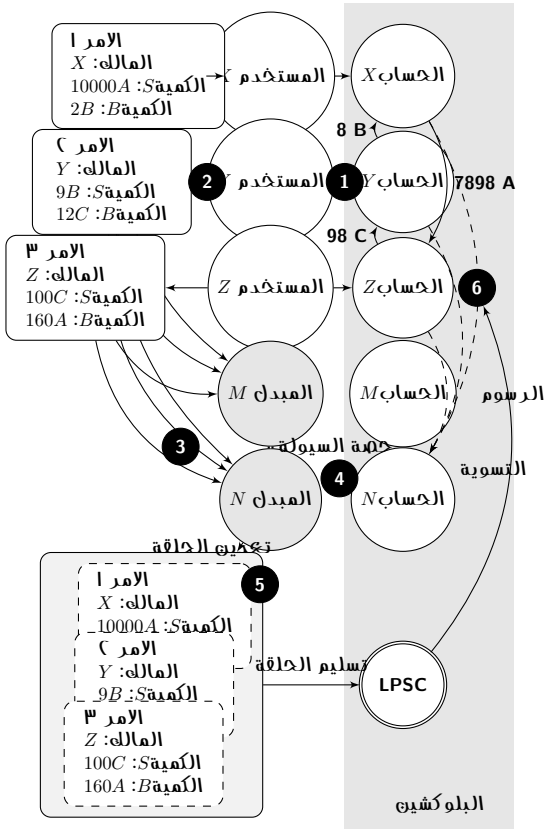
5. عملية منصات التداول

(٠). **ترخيص البروتوكول** : في الشكل ٢ ، المستخدم Y_{who} يريد تبداول العملات الرمزية ويأذن ل LPSC للتعامل مع عدد العملات الرمزيةB التي يريد المستخدم بيعها. لا يؤدي هذا إلى قفل العملات الرمزية للمستخدم ، الذي يظل حر في نقلها أثناء معالجة الطلب..

(٠). **إنشاء الطلب** : السعر الحالي وحجز الطلب للعملة الرمزية B مقابل العملة الرمزية C يتم توفيرها من خلال المبدلات أو العوامل الأخرى المرتبطة بالشبكة ، مثل متصفحات حجز الطلب. يضع المستخدم Y أمرًا (أمرًا محددًا) يحدد المبلغ S والمبلغ B والمعلمات الأخرى من خلال أي واجهة محفظة مدمجة. يمكن إضافة مبلغ من LR_x إلى الطلب كرسوم لمنقبين الحاقة ؛ رسوم أعلى ل LR_x تعني فرصة أفضل للمعالجة في وقت اسرع من قبل المنقبين. يتم توقيع تجزئة الطلب مع المفتاح الخاص للمستخدم Y.

(٠). **بث الطلب** : ترسل المحفظة الطلب وتوقيعه إلى واحد أو أكثر من المبدلات.المبدلات تحدث حجز الطلب العام لها. لا يحتاج البروتوكول إلى حجوزات الطلب لكي يتم بناؤها بطريقة معينة ، مثل من ياتي اولا يخدم اولا. بدلاً من ذلك ، تمتلك المرحلات القدرة على اتخاذ قرارات التصميم الخاصة بها في بناء حجوزات طلباتها.

(٠). **تقاسم السيولة** : تبث المبدلات الطلب إلى مرحلات (مبدلات) أخرى من خلال أي وسيلة اتصال عشوائية. مرة أخرى ، هناك مرونة كيف \ اي من العقد تتفاعل. ولتسهيل مستوى معين من الاتصال بالشبكة ، هناك شبكة الترحيل (التبديل) مدمجة لتقاسم السيولة باستخدام البلوكشين المجمع. كما ذكرنا في القسم السابق ، تم تحسين شبكة الترحيل هذه للسرعة والشمولية.



١.٥ شكل: عملية تبادل $loopring$

(٠). **تعدين الحلقة (مطابقة الطلب)** : يحاول المنقبين تنفيذ الطلب كليًا أو جزئيًا بسعر الصرف المحدد أو بشكل أفضل من خلال مطابقته مع طلبات أخرى متعددة. تعدين الحلقة هو السبب الرئيسي في أن البروتوكول قادر على توفير سيولة عالية على أي زوج. إذا كان المعدل الذي تم تنفيذه أفضل من ما يحدده المستخدم Y ، يتم مشاركة الهامش بين جميع الطلبات في حلقة الطلب. كمكافأة ، يختار منقب ا لحلقة بين المطالبة بجزء من الهامش (فصل الهامش(، وإعادة LR_x إلى المستخدم) ، أو ببساطة الاحتفاظ برسوم LR_x.

(٠). **التحقق والتسوية** : يتم استلام حلقة الطلب بواسطة LPSC . يقوم بإجراء العديد من الفحوصات للتحقق من البيانات التييتوفرها منقب الحلقة وتحدد ما إذا كان من الممكن تسوية حلقة الطلب كليًا أو جزئيًا (اعتمادًا على معدل تنفيذ الطلبات داخل الحلقة والعملات الرمزية في محافظ المستخدمين). في حالة نجاح جميع عمليات التحقق ، يقوم العقد بتحويل العملات الرمزية إلى المستخدمين ويدفع رسوم منقب الحلقة والمحفظة في نفس الوقت. إذا كان رصيد المستخدم Y كما هو محدد بواسطة LPSC غير كافٍ ، فسيتم اعتباره منخفض: الطلب المنخفض سيتم تلقائيًا تغييره إلى حجمه الأصلي إذا تم إيداع أموال كافية في عنوانه ، بخلاف الإلغاء ، الذي يعتبر عملية يدوية احادية الطريق ولا يمكن عكسها.

6. المرونة التشغيلية

من المهم ملاحظة أن معيار لوبرنج المفتوح يتيح للمشاركين مرونة كبيرة في كيفية عملهم. الفاعلون أحرار في تنفيذ نماذج أعمال جديدة وتوفير قيمة للمستخدمين ، وكسب رسوم LR_x على الحجم أو غيرها من المقاييس في العملية (إذا اختاروا ذلك). النظام نظامي ويهدف إلى دعم المشاركة من العديد من التطبيقات.

١.٦ سجل الطلبيات

يمكن للمرحلات تصميم سجلات طلباتها بأي عدد من الطرق لعرض ومطابقة طلبات المستخدمين. يتبع التنفيذ الأول لسجل الطلب الخاص بنا نموذج OTC ، حيث يتم وضع حد الأوامر على أساس السعر وحده. بعبارة أخرى ، لا تؤثر الطوابع الزمنية للأوامر على سجل الطلبات. ومع ذلك ، فإن المرحل حر في

تصميم سجل الطلبات الخاص به بطريقة تحاكي محرك المطابقة النموذجي للتبادل المركزي ، حيث يتم ترتيب الطلبات حسب السعر ، مع احترام الطوابع الزمنية أيضاً. إذا كان المرحل يميل إلى تقديم هذا النوع من سجل الطلبات ، فيمكنه امتلاك \ دمج مع المحفظة ، وإرسال أوامر المحفظة هذه فقط إلى المرحل المنفرد ، الذي سيكون قادراً على مطابقة الطلبات بناءً على الوقت. أي تكوين من هذا القبيل ممكن

في حين تتطلب بروتوكولات DEX الأخرى في بعض الأحيان المرحلات للحصول على موارد - الأرصدة الأولية للعملة الرمزية لوضع أوامر المستقبلين - تحتاج مرحلات لوبرنج فقط إلى العثور على أوامر قابلة للتطابق لكي يتم إتمام الصفقة ، ويمكنها القيام بذلك بدون العملات الرمزية الأولية.

٢.٦ تقاسم السيولة

المرحلات حرة في تصميم كيفية مشاركة السيولة (الطلبات) مع بعضها البعض. إن اتحاد البلوكشين الخاص بنا ليس سوى حل واحد لإنجاز ذلك ، والنظام حر في التواصل والاتصال كما يحلو له. وبالاضافة الى الانضمام الى اتحاد البلوكشين ، يمكنها بناء وإدارة شؤونها الخاصة ، ووضع القواعد \ الحوافز على النحو الذي يرونها مناسبا. يمكن أن تعمل المرحلات أيضاً بمفردها ، كما هو واضح في تنفيذ المحفظة الحساسة للوقت. بالطبع ، هناك مزايا واضحة في التواصل مع المرحلات الأخرى في السعي إلى مؤثرات الشبكة ، ومع ذلك ، قد تستلزم نماذج العملات المختلفة تصاميم مشاركة غريبة وتقسيم الرسوم بأي عدد من الطرق.

7. مواصفات البروتوكول

١.٧ تحليل الطلب

الطلب عبارة عن حزمة من البيانات التي تصف القصد من تداول المستخدم. يتم تعريف أمر لوبرنج باستخدام نموذج طلب أحادي الاتجاه ، أو UDOM ، كما يلي:

```
message Order {
address protocol;
address owner;
address tokenS;
address tokenB;
uint256 amountS;
uint256 amountB;
unit256 lrcFee
%unit256 validSince; // Seconds since epoch
%unit256 validUntil; // Seconds since epoch
%uint8 marginSplitPercentage; // [1-100]
bool buyNoMoreThanAmountB;
uint256 walletId;
%// Dual-Authoring address
address authAddr;
// v, r, s are parts of the signature
uint8 v;
bytes32 r;
bytes32 s;
%// Dual-Authoring private-key,
%// not used for calculating order's hash,
%// thus it is NOT signed.
string authKey;
}
```

لضمان أصل الطلب ، يتم توقيعه مقابل تجزئة معلماته ، باستثناء authAddr ، مع المفتاح الخاص للمستخدم. يتم استخدام المعلم authAddr لتوقيع حلقات الطلب التي يتم ترتيب هذا الطلب كجزء منها ، والذي يمنع التشغيل الأمامي. يرجى الرجوع إلى القسم ١.٩ لمزيد من التفاصيل. يتم تمثيل التوقيع بواسطة v ، r ، و $sfields$ ، ويتم إرسالها بجانب معلمات الطلب عبر الشبكة. وهذا يضمن

بقاء الطلب ثابتاًا خلال فترة حياته الكاملة. على الرغم من أن الطلب لن يتغير أبدًا ، لا يزال بإمكان البروتوكول حساب حالته الحالية استنادًا إلى رصيد عنوانه بالإضافة إلى متغيرات أخرى.

لا يشمل UDOM السعر (والذي يجب أن يكون رقماًا عشريا بطبيعته) ، ولكن بدلاً من ذلك يستخدم المصطلح معدل أو ر ، والذي يتم التعبير عنه كمقدار \ مبلغ. المعدل ليس رقم عشري بل هو تعبير سيتم تقييمه فقط مع أعداد صحيحة أخرى غير موقعة عند الطلب ، للحفاظ على جميع النتائج الوسيطة كأعداد صحيحة غير موقعة وزيادة دقة الحساب

١.١.٧ مبالغ الشراء

عندما يقوم منقب الحلقة بمطابقات حلقة الأوامر ، من الممكن أن يكون معدل أفضل قابلاً للتنفيذ ،يسمح للمستخدمين بالحصول على مزيد من العملات الرمزية B بمبلغ B الذي حددوه. ومع ذلك ، إذا كان buyNoMoreThanAmountB تم تعيينه على $True$ ، يضمن البروتوكول أن المستخدمين لا يتلقون أكثر من مبلغ B من العملات الرمزية B. وهكذا، يحدد المعلم buyNoMoreThanAmountB ل UDOM متى يعتبر الطلب منفذ كليا. buyNoMoreThanAmountB يطبق حد أقصى على المبلغض أو المبلغ B ، ويسمح للمستخدمين بالتعبير عن نوايا تجارية أكثر دقة من أوامر الشراء \ البيع التقليدية.

على سبيل المثال: مع $amountS = 10$ و $amountB = 2$ ، المعدل $r = \frac{10}{2} = 5$. وهكذا يكون المستخدم على استعداد لبيع ٥ عمل رمزية لكل عمل رمزية B. يطابق منقب الحلقة ويجد للمستخدم معدل ٤ ، مما يسمح للمستخدم بتلقي ٥.٢ عملة رمزية B بدلاً من ٢ . مع ذلك ، إذا كان المستخدم يريد ٢ عملة رمزية B فقط وقام بتعيين إشارة buyNoMoreThanAmountB إلى $True$ ، يقوم $LPSC$ بإجراء المعاملة بسعر ٤ ويبيع المستخدم ٤ عمل رمزية S لكل عملة رمزية B ، ويوفر بشكل فعال ٢ عمل رمزية S. ضع في اعتبارك أن هذا لا يأخذ في الاعتبار رسوم التعدين (انظر القسم ١.٨).

```
Order(amountS,tokenS,
amountB,tokenB,
buyNoMoreThantokenB)
```

لتمثيل طلب في شكل مبسط ، فانه بالنسبة لأسواق ETH/USD في منصات التداول التقليدي ، يمكن لنماذج البيع والشراء التقليدية التعبير عن الطلب الأول والثالث أدناه ، ولكن ليس الثاني:

(٠). بيع ٠١ ETH بسعر ٠٠٣ دولار أمريكي \ ETH. يمكن التعبير عن هذا الطلب على النحو التالي: Order(10, ETH, 3000, USD, False).

(٠). بيع ETH بسعر ٠٠٣ دولار أمريكي \ ETH للحصول على ٠٠٠٣ دولار أمريكي. يمكن التعبير عن هذا الطلب على النحو التالي: Order(10, ETH, 3000, USD, True).

(٠). اشتر ٠١ ETH بسعر ٠٠٣ دولار أمريكي \ ETH ، يمكن التعبير عن هذا الطلب على النحو التالي: Order(3000, USD, 10, ETH, True).

(٠). أنفق ٠٠٠٣ دولار أمريكي لشراء أكبر عدد ممكن من ETH بسعر ٠٠٣ دولار أمريكي \ ETH ، يمكن التعبير عن هذا الطلب على النحو التالي: Order(3000, USD, 10, ETH, False).

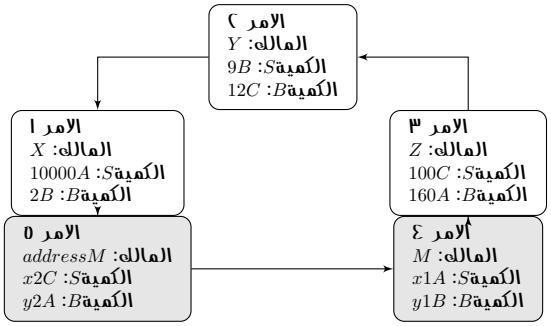
٢.٧ التحقق من الحلقة

لا تقوم عقود Loopring الذكية بإجراء عمليات حساب أو كمية سعر الصرف ، ولكن يجب أن تتلقى و تتحقق مما يزودها به منقبين الحلقة لهذه القيم. يتم إجراء هذه الحسابات من قبل منقبين الحلقة لسببين رئيسيين: (١) لغة البرمجة للعقود الذكية ، مثل الصلابة [8] على الاثريوم ، لا يوجد لديها دعم لرياضيات العدد العشري ، لا سيما اعداد القوى $pow(x; 1 = n)$ (حساب الجذر رقم n للرقم العشري) ، و (٢) من المستحسن أن يتم إجراء عملية حسابية خارج السلسلة للحد من حساب وتكلفة البلوكشين.

١.٢.٧ فحص فرع الحلقة

هذه الخطوة تمنع المراجعين من تحقيق جميع الهامش بشكل غير عادل في حلقة الطلب عن طريق تنفيذ أوامر جديدة داخله. وبشكل أساسي ، بمجرد العثور على حلقة طلب صالحة بواسطة منقب

الحلقة ، قد يكون من المغري إضافة أوامر أخرى إلى حلقة الطلب لاستيعاب هامش المستخدمين (خصومات السعر) بشكل كامل. كما هو موضح في الشكل ٣ أدناه ، فإن النتائج المحسوبة بدقة $x_1, y_1, x_2 \; and \; y_2$ ستجعل ناتج معدل جميع الطلبات هو ١ بالضبط ، وبالتالي لن يكون هناك خصم للأسعار.



١.٧ شكل: حلقة طلب مع فرعها

هذا هو خطرالصفر ، إضافة القيمة صفر إلى الشبكة ، ويعتبر السلوك غير عادل من قبل منقب الحلقة. لمنع هذا ، تتطلب Loopring أن الحلقة الصالحة لا يمكن أن تحتوي على أي حلقات فرعية. للتحقق من ذلك ، يضمن LPSC أن لا تكون العملة الرمزيه في موضع الشراء.

أو البيع مرتين. في الرسم البياني أعلاه ، يمكننا أن نرى أن العملة الرمزية A هي عملة رمزية للبيع مرتين وعملة رمزية للشراء مرتين ، والذي لن يتم السماح به.

٢.٢.٧ التحقق من تنفيذ السعر

يتم إجراء حسابات سعر الصرف في حلقة الطلب من قبل منقبين الحلقة لأسباب مذكورة أعلاه. يجب على LPSC التحقق من صحتها. أولاً ، تتحقق من أن معدل الشراء الذي يمكن لمنقب الحلقة تنفيذه لكل طلب يساوي أو أقل من سعر الشراء الأصلي الذي يحدده المستخدم. يضمن ذلك للمستخدم الحصول على الأقل على سعر الصرف الذي طلبه أو أفضل من الصفقة. وبمجرد تأكيد أسعار الصرف ، يضمن LPSC أن كل طلب في حلقة الطلب يشترك بنفس الخصم في السعر. على سبيل المثال ، إذا كان السعر المخفض هو Y ، فسيكون السعر لكل طلب: $r_{0\to1}\cdot(1-\gamma), r_{1\to2}\cdot(1-\gamma), r_{2\to0}\cdot(1-\gamma),$ والتي تحقق

$$r_{0\to1}\cdot(1-\gamma), r_{1\to2}\cdot(1-\gamma), r_{2\to0}\cdot(1-\gamma) = 1 \quad (٣)$$

وبالتالي:

$$\gamma = 1 - \frac{1}{\sqrt[3]{r_{0\to1}\cdot r_{1\to2}\cdot r_{2\to0}}} \quad (٤)$$

إذا تجاوزت المعاملة عدد الطلبات ، يكون الخصم:

$$\gamma = 1 - \frac{1}{\sqrt[n]{\prod_{i=0}^{n-1} r^i}} \quad (٥)$$

حيث r^i هو معدل دوران الطلب للطلب $i-th$. من الواضح ، فقط عندما يكون معدل الخصم هو $\gamma \geq 0$ ، يمكن تنفيذ هذه الأوامر ؛ وسعر الصرف الفعلي للأمر $i-th \; order(O^i)'s$ هو

$$\hat{r^i} = r^i \cdot (1-\gamma), \hat{r^i} \leq r^i \quad (٦)$$

٣.٢.٧ تعقب وإلغاء التنفيذ

يمكن للمستخدم إلغاء الطلب جزئياًا أو كلياًا عن طريق إرسال معاملة خاصة إلى LPSC ، تحتوي على تفاصيل حول الطلب والمبلغ المراد إلغاءه. يأخذ LPSC ذلك في الاعتبار ، يخزن المبلغ المراد إلغاءه ، ويبعث حدث OrderCancelled إلى الشبكة. يتتبع LPSC المبالغ التي تم تنفيذها وإلغائها عن طريق تخزين قيمها باستخدام تجزئة الطلب كمعرف. هذه البيانات يمكن الوصول إليها بشكل عام و الاحداث OrderCancelled/ OrderFilled تُصدر عند تغييرها. تتبع هذه القيم أمر بالغ الأهمية لLPSC خلال خطوة تسوية حلقة الطلب.

كما يدعم LPSC إلغاء جميع الطلبات الخاصة بأي زوج تداول مع الحدث OrdersCancelled ويلغي جميع الطلبات لعنوان مع حدث AllOrdersCancelled.

٤.٢.٧ قياس الطلب

يتم قياس الطلبات وفقاًا لتاريخ المبالغ المنفذة والملغاة والرصيد الحالي لحسابات المرسلين. تجد العملية الطلب مع أصغر مقدار يتم تنفيذة وفقاًا

للخصائص المذكورة أعلاه وتستخدمه كمرجع لقياس جميع المعاملات في حلقة الطلب.

يمكن أن يساعد العثور على الطلب ذو أدنى قيمة في معرفة حجم التنفيذ لكل طلب. على سبيل المثال ، إذا كان الطلب $i-th$ هو أقل قيمة ، فان عدد العملات الرمزية المباعة من كل طلب S و عدد العملات الرمزية المشتراة b من كل طلب يمكن حسابها على النحو التالي:

$$\hat{s^i} = \bar{s_i}, \hat{b^i} = \hat{s^i}/\hat{r^i};$$
$$s^{\hat{i\oplus 1}} = \hat{b^i}, b^{\hat{i\oplus 1}} = s^{\hat{i\oplus 1}}/r^{\hat{i\oplus 1}};$$
$$s^{\hat{i\oplus 2}} = b^{\hat{i\oplus 1}}, b^{\hat{i\oplus 2}} = s^{\hat{i\oplus 2}}/r^{\hat{i\oplus 2}};$$
$$\dots$$

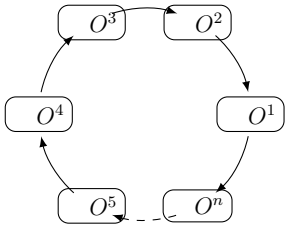
حيث s_i هو الرصيد المتبقي بعد تنفيذ الطلبات جزئياً.

أثناء التنفيذ ، يمكننا أن نفترض بأمان أي طلب في حلقة الطلب بأقل قيمة ، ثم يقوم بالتكرار عبر حلقة الطلب على الأكثر مرتين لحساب حجم التنفيذ لكل أمر.

مثال: إذا كان الحد الأدنى المطلوب تنفيذة مقارنة بالطلب الأصلي هو 5% ، فإن جميع المعاملات في حلقة الطلب يتم تخفيضها إلى 5%. بمجرد الانتهاء من المعاملات ، يجب التنفيذ كليا للطلب الذي كان يعتبر ذو أصغر كمية متبقية.

٣.٧ تسوية الحلقة

إذا كانت حلقة الطلب تلبي جميع الفحوصات السابقة ، يمكن إغلاق حلقة الطلب ، ويمكن إجراء المعاملات. هذا يعني أن جميع الطلبات n تشكل حلقة طلب مغلقة ، متصلة كما هو موضح في الشكل٤:



٢.٧ شكل: تسوية الحلقة

لإجراء المعاملات ، يستخدم LPSC العقد الذكي TokenTransferDelegate. إن إدخال مثل هذا التفويض يجعل تحديث بروتوكول العقد الذكي أسهل حيث أن جميع الطلبات تحتاج فقط إلى تصريح هذا التفويض بدلاً من إصدارات مختلفة من البروتوكول.

لكل طلب في حلقة الطلب ، يتم دفع العملات الرمزية إلى الطلب التالي أو السابق بناءً على التنفيذ. ثم يتم دفع رسوم منقب الحلقة حسب نموذج الرسوم الذي يختاره منقب الحلقة. وأخيراً ، بمجرد إجراء جميع المعاملات ، يتم إصدار حدث RingMined.

١.٣.٧ قياس الطلب

يُصدر البروتوكول أحداثاً تسمح للمرحلات و متصفحات الاوامر والممثلين الآخرين بتلقي تحديثات سجل الطلبات بأكبر قدر ممكن من الكفاءة. الأحداث المصدرة هي:

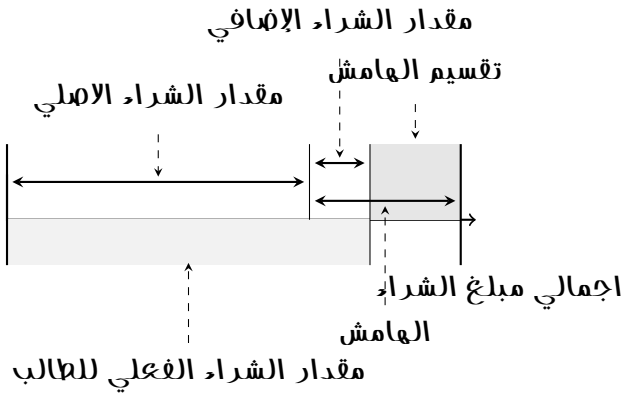
- OrderCancelled : تم إلغاء طلب معين.
- OrdersCancelled : تم إلغاء جميع طلبات زوج التداول من عنوان المالك.
- AllOrdersCancelled : تم إلغاء جميع طلبات جميع أزواج التداول من عنوان المالك.
- RingMined : تمت تسوية حلقة الطلب بنجاح. يحتوي هذا الحدث على بيانات متعلقة بكل نقل للعملة الرمزية داخل الحلقة.

8. العملة الرمزية LRx

LRx هي رمز العملة الرمزية العامة لدينا. LRC هي العملة الرمزية ل شرذ على اثريوم ، LRQ على $Qtum$ ، و LRN على NEO ، إلخ. سيتم إدخال أنواع LRx الأخرى في المستقبل حيث يتم نشر Loopring على تقنيات البلوكشين العامة الأخرى.

١.٨ نموذج الرسوم

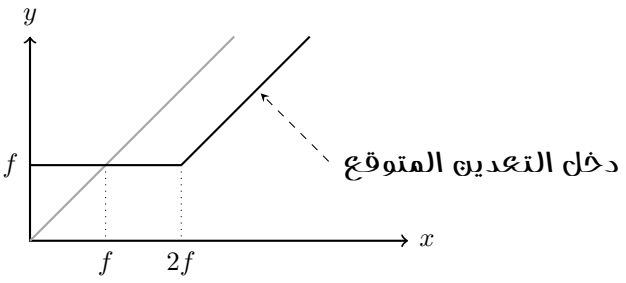
عندما يقوم المستخدم بإنشاء طلب ، فإنه يحدد مبلغ LRx ليتم دفعه إلى منقب الحلقة كرسوم ، بالاقتران مع نسبة الهامش تستش(مرنضلتصرنتَ) الذي يتم إجراؤه بناء على الطلب الذي يمكن لمنقب الحلقة المطالبة به. وهذا ما يسمى بتقسيم الهامش. يتم ترك قرار أي واحد للاختيار (الرسوم أو تقسيم الهامش) إلى منقب الحلقة . تمثيل تقسيم الهامش:



١.٨ شكل: هامش بنسبة ستين بالمية

إذا كان الهامش الموجود على حلقة الطلب صغيراً جداً ، فسيقوم منقب الحلقة باختيار رسوم LRx. إذا كان الهامش ، على النقيض من ذلك ، الهامش يكون كبيراً بما فيه الكفاية لانتاج تقسيم الهامش الذي قيمتة أكثر بكثير من رسوم LRx ، سيختار منقب الحلقة تقسيم الهامش. هناك شرط آخر ، على أي حال: عندما يختار منقب الحلقة تقسيم الهامش ، يجب أن يدفع للمستخدم (منشئ الطلب) رسوم ، و تساوي LRx التي كان المستخدم سيدفعها إلى منقب الحلقة كرسوم. هذا يزيد القيمة الحدية حيث سيختار منقب الحلقة تقسيم الهامش إلى ضعف رسوم LRx للطلب ، مما يزيد من الميل الى خيار رسوم LRx. يسمح هذا لمنقب الحلقة بالحصول على دخل ثابت على حلقات الطلب ذات الهامش المنخفض لمقايضة تلقي دخل أقل في حلقات الطلب ذات الهامش الأعلى. يعتمد نموذج الرسوم لدينا على التوقع بأنه مع نمو السوق ونضوجه ، سيكون هناك عدد أقل من حلقات الطلب ذات الهامش المرتفع ، مما يتطلب رسوم LRx ثابتة كحافز.

ننتهي بالرسم البياني التالي:



٢.٨ شكل: نموذج رسوم Loopring

حيث f هي الرسوم ، x هو تقسيم الهامش ، y هو دخل التعدين. $y = max(f, x - f)$ كما هو مشار إليه بواسطة الخط الصلب ؛ إذا كانت رسوم LRx للطلب ٠ ، فإن المعادلة تكون $y = max(0, x - 0)$ والتي تبسط إلى $y = x$ كما هو محدد بالخط الرمادي.

النتائج هي:

(٠). اذا كان تقسيم الهامش ٠ ، سوف يختار منقبين الحلقة رسوم LRx السطحية ولا يزالون محفزين.

(٠). إذا كانت رسوم LRx هي ٠ ، فإن نتائج الخط الرمادي والدخل يستند إلى نموذج الخط العام.

(٠). عندما يكون دخل تقسيم الهامش أكبر من $x2$ (رسوم LRx) ، يختار منقبين الحلقة تقسيم الهامش ويدفعوا LRx إلى المستخدم.

تجدر الإشارة إلى أنه إذا كانت رسوم LRx غير صفرية ، وبغض النظر عن الخيار الذي يختاره منقب الحلقة ، فسيكون هناك دائماً نقل ل LRx بين منقب الحلقة ومرسل الطلب. إما أن يحصل منقب الحلقة على رسوم LRx ، أو يدفع رسوم LRx مرة أخرى إلى المرسل لاتخاذ تقسيم الهامش.

سوف يشارك منقبين الحلقة بنسب معينة من الرسوم مع المحافظ. عندما يضع المستخدم طلباً عبر محفظة ويتم تنفيذة ، تتم مكافأة المحفظة بجزء من الرسوم أو تقسيم الهامش. على الرغم من أن هذه الوحدات ، ونماذج الأعمال الفريدة أو التطبيقات ممكنة ، إلا أن ميلنا هو أن تستلم المحافظ ما يقارب $20\% - 25\%$ من الرسوم المكتسبة. تمثل المحافظ هدفاً أساسياً لتكامل بروتوكول Loopring نظراً لأنها تحتوي على قاعدة المستخدمين ، ولكن مصدر الدخل قليل أو معدوم.

٢.٨ نظام العمل اللامركزي

في الاصل ، بروتوكول Loopring هو بروتوكول اجتماعي بمعنى أنه يعتمد على التنسيق بين الأعضاء للعمل بفعالية نحو الهدف. وهذا لا يختلف عن بروتوكولات التشفير الاقتصادية بشكل عام ، بل إن فائدتها محمية إلى حد كبير من خلال نفس آليات مشاكل التنسيق [6] ، واتزان الحدث القاسي ، والعقلانية المحدودة. ولهذه الغاية ، لا تستخدم العملات الرمزية LRx فقط لدفع الرسوم ، ولكن أيضاً لمواءمة الحوافز المالية للمشاركين في الشبكة المختلفة. ومثل هذا المواءمة ضروري لاعتماد واسع النطاق لأي بروتوكول ، ولكنه شديد الحدة بالنسبة لبروتوكولات منصات التداول ، بالنظر إلى أن النجاح يعتمد بشكل كبير على تحسين السيولة في نظام لامركزي قوي.

سيتم استخدام العملات الرمزية LRx لتفعيل تحديثات البروتوكول من خلال الإدارة اللامركزية. تخضع تحديثات العقود الذكية لحاملي العملة الرمزية لضمان الاستمرارية والسلامة ، وللتخفيف من مخاطر سحب السيولة من خلال عدم التوافق. وبالنظر إلى أن العقود الذكية لا يمكن تغييرها بمجرد نشرها ، فهناك خطر بأن تستمر ال $dApps$ أو المستخدمين النهائيين في التفاعل مع الإصدارات التي تم إيقافها وإبطال نظرهم من العقود المحدثة .

تعتبر قابلية الترقية أمراً حاسماً لنجاح البروتوكول حيث أنه يجب أن يتكيف مع متطلبات السوق و تقنيات البلوكشين الكامنة. ستسمح الإدارة اللامركزية من قبل أصحاب المصلحة في LRx بتحديثات بروتوكول العقد الذكي دون تعطيل ال $dApps$ أو المستخدمين النهائيين ، أو الاعتماد بشكل كبير على تجريد العقود الذكية. في البداية ، سوف يتم ذلك من خلال عقد ذكي بسيط متعدد التوقيعات ، بهدف التقدم نحو نوع آلية DAO.

9. الحمايه من الاحتيال و الهجوم

١.٩ الوقاية من التداول المسبق

في منصات التداول ات اللامركزية ، يتم التداول المسبق عندما يحاول شخص ما نسخ حل التداول لعقدة آخرى ، ويجعله مقوضاً قبل المعاملة الأصلية الموجودة في تجمع المعاملات المعلق (mempool). ويمكن تحقيق ذلك عن طريق تحديد رسوم معاملات أعلى (سعر الجاز). المخطط الرئيسي للتداول المسبق في Loopring (وأية بروتوكولات لمطابقة الطلب) هي عبارة عن سرقة الطلب:

عندما يسرق القائم بالتداول المسبق أمراً واحداً أو أكثر من معاملة تسوية طلبات الحلقة المعلقة ؛ وبالنسبة الى Loopring : عندما يسرق القائم بالتداول المسبق حلقة الطلب بالكامل من معاملة معلقة.

عندما لا يتم تأكيد معاملة submitRing ولا تزال

في تجمع المعاملات المعلقة ، يمكن لأي شخص أن يكتشف مثل هذه المعاملة بسهولة ويستبدل عنوان المنقب بعنوانه الخاص (filcherAddress) ، ثم يمكنه إعادة التوقيع على الحمولة باستخدام filcherAddress لاستبدال توقيع حلقة الطلب. يمكن أن يقوم المختلس بتعيين سعر أعلى للجاز وتقديم معاملة جديدة على أمل أن يقوم منقبين الكتلة باختيار معاملته الجديدة في الكتلة التالية بدلاً من المعاملة الأصلية submitRing.

الحلول السابقة لهذه المشكلة كانت لها جوانب سلبية مهمة: تتطلب المزيد من المعاملات وبالتالي تكلف منقبين الحلقة الكثير من الجاز ؛ وتأخذ على الأقل ضعف الكتل لتسوية حلقة الطلب. حلنا الجديد ، التأليف المزدوج [19] ، ينطوي على آلية إنشاء مستويين من الترخيص للأوامر - واحد للتسوية ، والآخر للتنقيب عن الحلقة.

عملية التأليف المزدوج:

(٠). لكل طلب ، سيولد برنامج المحفظة زوجاً عشوائياً من المفتاح العام \ المفتاح الخاص ، ويضع زوج المفاتيح في مقتطف $JSON$ التابع للطلب. (البديل هو استخدام العنوان المستمد من المفتاح العام بدلاً من المفتاح العام نفسه لتقليل حجم البايت. نستخدم authAddr لتمثيل مثل هذا العنوان ، و authKeys لتمثيل المفتاح الخاص المطابق لـ authAddr).

(٠). حساب تجزئة الطلب مع جميع الحقول في الطلب باستثناء (r و v و s و authKeys) ، وتوقيع التجزئة باستخدام المفتاح الخاص بالمالك (وليس authKeys).

(٠). سترسل المحفظة الطلب مع authKeys إلى المرحلات من أجل تعدين الحلقة. سيتحقق منقبين الحلقة من أن authKeys و authAddr قد تم إقرانها بشكل صحيح وأن توقيع الطلب صالح فيما يتعلق بعنوان المالك.

(٠). عند تحديد حلقة الطلب ، سيستخدم منقب الحلقة authKey لكل طلب لتوقيع تجزئة الحلقة وminerAddress وجميع معلمات التعدين. إذا كانت حلقة الطلب تحتوي على n من الأوامر ، فستكون هناك توقيعات n بواسطة n من ال authKeys . نحن نسمي هذه التواقيع ب authSignatures .قد يحتاج منقب الحلقة أيضاً إلى توقيع تجزئة الحلقة مع جميع معلمات التعدين باستخدام المفتاح الخاص لـ minerAddress.

(٠). يستدعي منقب الحلقة دالة submitRing مع كافة المعلمات ، بالإضافة إلى authSignature الإضافية. لاحظ أن authKeys ليست جزءاً من المعاملة على السلسلة ، وبالتالي تظل غير معروفة لأطراف أخرى غيرمنقب الحلقة نفسه.

(٠). سيقوم بروتوكول Loopring الآن بالتحقق من صحة كل من authSignatureagainst مقابل authAddr لكل طلب ، ويرفض حلقة الطلب إذا كان أي authSignature مفقودة أو غير صالح.

والنتيجة هي الآن:

- يضمن توقيع الطلب (بواسطة المفتاح الخاص لعنوان المالك) بعدم إمكانية تعديل الطلب ، بما في ذلك authAddr.
- توقيع منقب الحلقة (بواسطة المفتاح الخاص ب minerAddress) ، إذا تم توفيره ، يضمن أنه لا يمكن لأي شخص استخدام هويته للتنقيب عن حلقة الطلب.
- يضمن authSignatures لا يمكن تعديل حلقة النظام بأكملها ، بما في ذلك minerAddress، ولا يمكن سرقة أي أوامر.

يمنع التأليف المزدوج اختلاس الحلقة واختلاس الطلب مع ضمان استمرار تسوية حلقات الطلب في معاملة واحدة. بالإضافة إلى ذلك ، تفتح ميزة التأليف المزدوج أبواباً أمام المرحلات لمشاركة الطلبات بطريقتين: المشاركة الغير قابلة للمطابقة والمشاركة القابلة للمطابقة. بشكل افتراضي ، تقوم Loopring بتشغيل نموذج OTC وتدعم فقط طلبات السعر المحدد ، بمعنى أنه يتم تجاهل الطوابع الزمنية للأوامر. ويعني هذا أن تداول التداول المسبق لا يؤثر على السعر الفعلي لذلك التداول ، ولكنه يؤثر على ما إذا تم تنفيذه أم لا.

10. الهجمات الأخرى

١.١٠ هجوم *Sybil* او *DOS*

يمكن للمستخدمين الضارين - بصفتهم أنفسهم أو بهويات مزورة - إرسال عدد كبير من الطلبات الصغيرة لمهاجمة عقد Loopring . ومع ذلك ، نظرًا لأننا نسمح للعقد برفض الطلبات استنادًا إلى معاييرها الخاصة - والتي قد تخفي أو تكشف عنها - فإن معظم هذه الطلبات سيتم رفضها لعدم تحقيق أرباح مرضية عند مطابقتها. من خلال تمكين المرحلات لإملاء كيفية إدارة الطلبيات ، لا نرى هجومًا كبيرًا جدًا على النظام كتهديد.

٢.١٠ عدم كفاية الرصيد

يمكن للمستخدمين الضارين تسجيل وتوزيع الطلبات التي تكون قيمتها غير صفرية ، ولكن عنوانها لا يحتوي على رصيد في الواقع. يمكن للعقد أن تراقب وتلاحظ أن بعض الطلبات رصيدها الفعلي هو صفر ، وتحدث حالات الطلب وفقا لذلك ، ثم تتجاهلها. يجب أن تقضي العقد الوقت لتحديث حالة الطلب ، ولكن يمكنها أيضًا اختيار تقليل الجهد من خلال ، على سبيل المثال ، وضع العناوين في القائمة السوداء وإفلات الطلبات ذات الصلة.

11. الخلاصة

يبدأ بروتوكول Loopring كطبقة أساسية للتبادل اللامركزي. وبذلك ، فإن له تداعيات عميقة في كيفية تبادل الناس للأصول والقيمة. المال ، كسلعة وسيطة ، تسهل أو تحل محل تبادل المقايضة وتحل المصادفة المزدوجة لمشكلة الاحتياجات [17] ، حيث يجب أن يرغب اثنان من الأطراف المقابلة في بضاعة أو خدمة كل منهما. وبالمثل ، يهدف بروتوكول Loopring إلى الاستغناء عن اعتمادنا على مصادفة الرغبات في أزواج التداول ، وذلك باستخدام مطابقة الحلقة لتداولات أكثر سهولة. وهذا مفيد لكيفية تبادل المجتمع والأسواق للعمل الرمزية ، والأصول التقليدية ، وما وراء ذلك. في الواقع ، مثلما العملات المشفرة اللامركزية تشكل تهديد لسيطرة دولة ما على المال ، فإن البروتوكول التوافقي الذي يمكن أن يربط المتداولين (المستهلكين \ المنتجين) على نطاق واسع ، يشكل تهديدًا نظريًا لمفهوم المال نفسه.

تشمل مزايا البروتوكول ما يلي:

- عدم تضمن إدارة الطلب خارج السلسلة و التسوية على السلسلة أي تضحية في الأداء من أجل الأمن
- زيادة السيولة بسبب تعدين الحلقة وتقاسم الطلبات.
- يحل التأليف المزدوج المشكلة الخبيثة في التداول المسبق التي يواجهها جميع ال $DEXs$ ومستخدميها اليوم.
- تتيح العقود الذكية العامة المجانية لأي $dApp$ ببناء أو التفاعل مع البروتوكول.
- يسمح التوحيد القياسي بين المشغلين بتأثيرات الشبكة وتحسين تجربة المستخدم النهائي.

- تبقى الشبكة مرنة في تشغيل سجلات الطلب والتواصل.
- انخفاض العوائق التي تحول دون الدخول تعني انخفاض تكاليف العقد المرتبطة بالشبكة والمستخدمين النهائيين.
- التداول مجهول مباشرة من محافظ المستخدم.

12. شكر وتقدير

نود أن نعرب عن امتناننا لمرشدينا ومستشارينا وللعديد من الناس في المجتمع الذين كانوا مرتاحين وسخيين بمعرفتهم. على وجه الخصوص ، نود أن نشكر شو باي (من ChinaLedger) ؛ البروفيسور هايبين كان، اليكس تشنغ ، هونغفي دا ؛ يين كاو، شياو تشوان وو، تشن وانغ ،وى يو، نيان دوان، جون شياو، جيانغ تشيان، جيانجزو شيانغ، يبينج جوه، داهاي لي، كلفن لونج، هواشيا شيا، جون ما، و انسيفالو باث لمراجعة وتقديم تعليقات حول هذا المشروع.

Bibliography

[1] Bancor protocol. *URL https://bancor.network/*, 2017.

[2] Rossella Agliardi and Ramazan Gençay. Hedging through a limit order book with varying liquidity. 2014.

[3] Viktor Atterlönn. A distributed ledger for gamification of pro-bono time, 2018.

[4] Iddo Bentov and Lorenz Breidenbach. The cost of decentralization. `http://hackingdistributed.com/2017/08/13/cost-of-decent/`, Accessed: 2018-03-05.

[5] Vitalik Buterin. Ethereum: a next generation smart contract and decentralized application platform (2013). *URL* {*http://ethereum. org/ethereum. html*}, 2017.

[6] Vitalik Buterin. Notes on blockchain governance. `https://vitalik.ca/general/2017/12/17/voting.html`, Accessed: 2018-03-05.

[7] Patrick Dai, Neil Mahi, Jordan Earls, and Alex Norta. Smart-contract value-transfer protocols on a distributed mobile application platform. *URL: https://qtum. org/uploads/-files/cf6d69348ca50dd985b60425ccf282f3. pdf*, 2017.

[8] Chris Dannen. *Introducing Ethereum and Solidity*. Springer, 2017.

[9] Hernando de Soto. *The Mystery Of Capital*. Basic Books, 2000.

[10] Fortune. How to steal $500 million in cryptocurrency. `http://fortune.com/2018/01/31/coincheck-hack-how`, Accessed: 2018-03-30.

[11] Yaron Velner Loi Luu. Kybernetwork: A trustless decentralized exchange and payment service. `https://kr.kyber.network/assets/KyberNetworkWhitepaper.pdf`, Accessed: 2018-03-05.

[12] Robert McMillan. The inside story of mt. gox, bitcoin's 460 dollar million disaster. 2014.

[13] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. 2008.

[14] Sylvain Ribes. Chasing fake volume: a crypto-plague. `https://medium.com/@sylvainartplayribes/chasing-fake-volume-a-crypto-plague-ea1a3c1e0b5e`, Accessed: 2018-03-10.

[15] Fabian Schuh and Daniel Larimer. Bitshares 2.0: Financial smart contract platform, 2015.

[16] Supersymmetry. Remarks on loopring. `https://docs.loopring.org/pdf/supersimmetry-loopring-remark.pdf`, Accessed: 2018-03-05.

[17] Nick Szabo. Menger on money: right and wrong. `http://unenumerated.blogspot.ca/2006/06/menger-on-money-right-and-wrong.html`, Accessed: 2018-03-05.

[18] Fabian Vogelsteller. Erc: Token standard. *URL https://github.com /ethereum /EIPs /issues /20*, 2015.

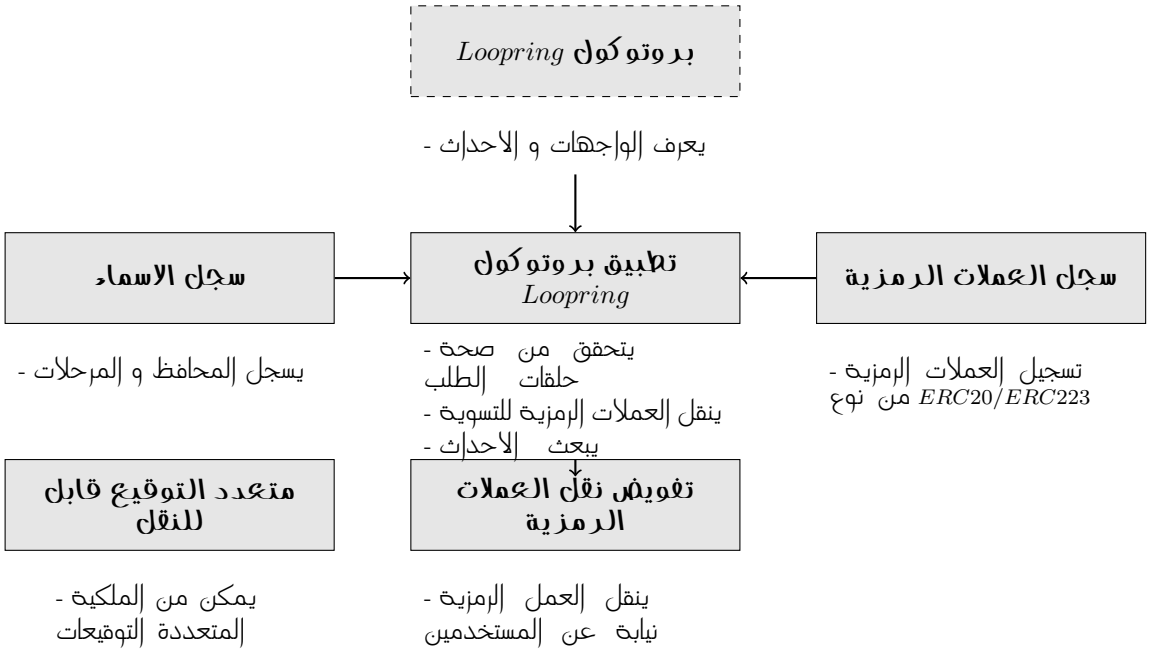
[19] Daniel Wang. Dual authoring — loopring's solution to front-running. *URL https://medium.com/loopring-protocol/dual-authoring-looprings-solution-to-front-running-d0fc9c348ef1*, 2018.

[20] Daniel Wang. Coinport's implemenation of udom. `https://github.com/dong77/backcore/blob/master/coinex/coinex-backend/src/main/scala/com/coinport/coinex/markets/MarketManager.scala`, Accessed: 2018-03-05.

[21] Will Warren and Amir Bandeali. 0x: An open protocol for decentralized exchange on the ethereum blockchain, 2017.

[22] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, 151, 2014.

١. *Loopring* على الاثريوم



١.١ شكل: العقود الذكية

ب. التطبيق

١.٢ الايثيريوم

تم تطبيق العقود الذكية التالية على الشبكة الرئيسية ل ايثريوم:

- LRC: $0xEF68e7C694F40c8202821eDF525dE3782458639f$
- $TokenRegistry$: $0xa21c1f2AE7f721aE77b1204A4f0811c642638da9$
- $TokenTransferDelegate$: $0x7b126ab811f278f288bf1d62d47334351dA20d1d$
- $NameRegistry$: $0xd181c1808e3f010F0F0aABc6Fe1bcE2025DB7Bb7$
- $LoopringProtocolImpl$: $0x0B48b747436f10c846696e889e66425e05CD740f$

٢.٢ كيوتم

تم تطبيق العقود الذكية التالية على الشبكة الرئيسية ل كيوتم:

- LRQ: $2eb2a66afd4e465fb06d8b71f30fb1b93e18788d$
- $TokenRegistry$: $c89ea34360258917daf3655f8bec555092350 9b3$
- $TokenTransferDelegate$: $60b3fa7f461664e4dafb621a36ac2722cc680f10$
- $NameRegistry$: $e26a27d92181069b25bc7283e03722f6ce7678bb$
- $LoopringProtocolImpl$: $5180bb56b696d16635abd8dc235e0ee432abf25d$