ReFX: 型に基づくスマートコントラクト自動検証器

ReFX: A type-based automated verifier for smart contracts

陳然・齋藤大聖・河田旺・西田雄気・五十嵐淳・末永幸平（京都大学情報学研究科）
古瀬淳（ダイラムダ株式会社）

課題：スマートコントラクトの安全性

最近のブロックチェーンにはスマートコントラクトというプログラムを動かす仕組みがあり、取引の自動化などに使われている。しかし、スマートコントラクトにバグがあると、暗号通貨が盗まれ、莫大な金銭的損失が生まれてしまうことがある。本研究では、プログラムが仕様を満たしていることを検査する形式検証手法を提案する。

本研究の内容

ブロックチェーン “Tezos” で動作する スマートコントラクトの自動検証手法

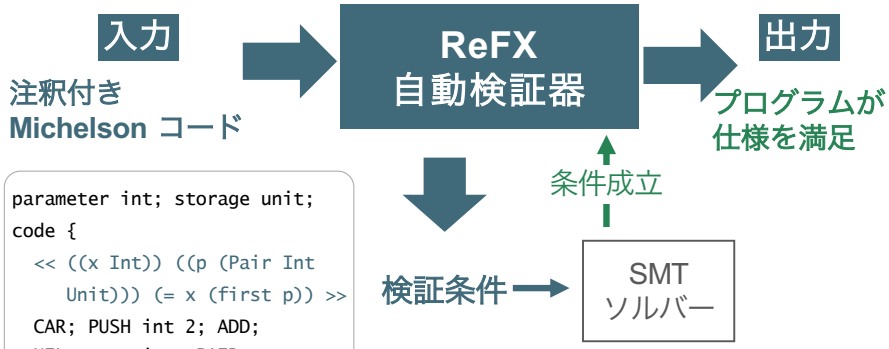
「ブロック」というデータ単位が鎖のように連結するデータベースである。P2Pネットワークによる分散的保存及び自律的な管理が特徴である。

Proof-of-Stake 合意モデルに基づいた第3世代ブロックチェーン技術の暗号通貨である。プロトコルが自身を修正でき、形式検証を念頭に設計されている。

ブロックチェーン上のアカウントに付随するプログラムである。送金されると自動的に実行され、プログラム中で再び送金を引き起こし、第三者を介さずにトランザクションを処理できる。

ReFX の構造

本研究は、Tezos のスマートコントラクトを書くプログラム言語 **Michelson** に対して、形式検証の手法で **ReFX 自動検証器**を開発した。Michelson コードに仕様を表現する注釈を加えて ReFX 自動検証器に入力すると検証条件を生成する。検証条件は SMT ソルバーに推論され、プログラムが仕様を満足するかどうかを判別できる。




```
parameter int; storage unit;
code {
  << ((x Int)) ((p (Pair Int
     Unit))) (= x (first p)) >>
  CAR; PUSH int 2; ADD;
  NIL operation; PAIR;
  << ((x Int)) ((p (Pair (List
    Operation) Int)))
    (= (+ x 2) (second p)) >>
}
```

例は一つの整数を受け取り、2 を加算して呼び先を返すとのスマートコントラクトである。<< >> の部分が仕様でコードを実行前のスタックの状態と終了後の状態を限定した。

Michelson

Tezos のスマートコントラクトを書くプログラム言語 Michelson は高レベルのデータ型とプリミティブを備え、厳密な静的型検査を導入したスタックベースのプログラム言語である。Michelson プログラムはシステムに導入する前に型検査が実行されるので、想定外のスタックでの実行などの理由で実行が失敗することがない。本研究では、Michelson の型システムに篩型を導入し、元の型検査器よりもっと安全性が高い型システムに拡張した。

篩型の導入

篩型は要素の基本型と属性を関連つけられ、値の範囲など、属性を限定を限定した型である。
Michelson はスタックを基いた言語なので、その型システムに適用する篩型を以下のような構文で記述できる：

$$\{(x_1 : T_1) : \ldots : (x_n : T_n) \mid \varphi\}$$

これは述語 φ を満足し、型 $T_1, \ldots, T_n$ を持つ要素 $x_1, \ldots, x_n$ でできた全てのスタックの型を表す。
述語は各要素の性質をもちろん、要素間の関係も表現できる。適切な述語を想定すれば、開発仕様を検証することや言語について特にバグを防げる性質を証明することができる。