平成24年度 情報工学コース卒業研究報告要旨

結縁 研究室	氏　　名	村 瀬　晃 弘
卒業研究題目	依存型を用いた高階関数型プログラム検証器の多相型への拡張	

本研究では，依存型を用いた高階関数型プログラム検証器の型システムを ML の多相型に拡張する．

関数型言語には，手続き型言語と比べて，プログラマの生産性とソフトウェアの信頼性を改善する多くの特徴がある．最も重要であるのは，強い静的型付けと型推論などの型システムである．関数型言語の型システムの有用性は，静的な型からコンパイル時に型を推論する能力から生じている。しかし，古典的な型システムでは，変数の値の詳細な情報を推論することができない。例えば，変数の値が整数であると推論することができても，変数の値が 0 以上で 10 より小さい，といった情報を推論することはできない．そのため，古典的な型システムは，整数 i で配列のインデックスの範囲外にアクセスするといった危険な操作の安全性を保証できない．そこで，依存型を用いて型付けと型推論を行う．

依存型とは，項に依存できる型である．依存型を用いることによって，プログラム中の満たすべき性質を型に記述でき，型検査や型推論でその性質を満たしているかどうかを検証できる．検証の結果，プログラムの実行前にエラーの種類や場所を検出できる．

依存型を用いた高階関数型プログラムに対する検証器として Depcegar がある．Depcegar では，refinement 型を使用して型推論を行う．Refinement 型とは，プログラムの動作を精密に表現するために一階述語論理を記述した依存型である．Depcegar では，型推論を補助する依存型の集合をあらかじめ定義することを必要とせず，CEGAR アルゴリズムを型推論に応用している．CEGAR アルゴリズムは，反例によってモデルを抽象洗練するアルゴリズムである．実際のモデルを過度に抽象化し，抽象モデルで与えられた性質を満たすのであれば，実際のモデルでも性質を満たすと結論付けることができる．性質を満たさなければ，検出された反例が本来実行可能でない反例かを検証する．もし反例が実際のモデルでも存在するものであれば，ループを終了する．そうでなければ，誤った反例をなくすように抽象モデルを洗練する．これらの段階を繰り返し，正しい結果を出力する．Depcegar では，型の候補を洗練していきプログラムを十分に型付けできる型を推論し，プログラムの安全性を検証している．

本研究では，Depcegar の型システムを ML などで使われている let 多相を用いた型システムに拡張し，拡張した型システムの健全性を証明する．let 多相では，let で束縛される値の型を推論し，具体化されなかった型変数に任意の型を代入することができる．let 多相を用いることで，同じ操作を適用する型ごとに定義する必要がなくなり，操作の定義を修正する必要がある場合に修正する箇所が減るので，プログラムの保守性を高めることができる．

本論文では，let 多相を用いて拡張した Depcegar の型システムを定義する．また，拡張した型システムの健全性を証明することにより，正しく型付けできればプログラムは表明違反にならないことを示す．