

# XHTML 構文検証手法における スクリプト要素の静的解析アルゴリズム

早瀬 康裕  
Yasuhiro Hayase

鷲尾 和則  
Kazunori Washio

松下 誠  
Makoto Matsushita

井上 克郎  
Katsuro Inoue

大阪大学大学院情報科学研究科コンピュータサイエンス専攻  
Department of Computer Science, Graduate School of Information Science and Technology, Osaka University

## 1 はじめに

Web で用いられている XHTML の文書構造は, Document Type Definition (DTD) に従うことが求められる. XHTML では ECMAScript[1] などのスクリプトにより, 文書内容が動的に生成される事が多いが, 既存の XHTML 検証手法では, 動的に生成される部分を無視して検証を行っていた. ところが, 静的には正しい XHTML だが, 動的に生成される部分が DTD に違反することがありうる. したがって, 動的に生成される部分も考慮して, XHTML 検証は行われなければならない.

そこで, 我々の研究グループでは, ECMAScript プログラムを静的に解析して, 出力される内容を考慮した XHTML 構文解析手法の提案 [2] を行っている. 本稿では, 本提案手法で用いられる, ECMAScript プログラム解析アルゴリズムの重要な要素である, 変数値の候補を計算するアルゴリズムについて述べる.

## 2 変数値候補の計算アルゴリズム

本アルゴリズムは, いくつかの制約を加えた ECMAScript プログラムを対象として, 任意の文中におけるある変数の取りうる値の候補を計算するものである.

本手法では最初に, ソースコード (図 1) から, フロー図 (図 2) を構築する. フロー図の頂点は文または式であり, 有向辺は始点の次に終点が行われることを示す. 値を調べたい文に対応する頂点を対象頂点, 値を調べたい変数を対象変数と呼ぶ.

対象頂点を始点とし, 対象変数への代入が見付かるまで, 有向辺を逆向きに辿る. 見付かった代入文の右辺に変数がある場合は, その変数の値を再帰的に調べ, 右辺の取り得る値を決定する. この手順だけでは, フロー図に回路がある場合に, 無限に再帰してしまう場合がある. これを避けるために, 調査中の頂点と変数の組を記憶しておき, 二重の探索を避ける.

以下, 対象頂点  $n$  を実行した直後の対象変数  $x$  の値の候補を  $x^n$  と表す. 値の候補は, 具体的な値 (1, "str", true 等) と, 型の決まった不定値 (不定値 (数値), 不定値 (文字列) 等) の集合で示される. 探索中の対象頂点と対象変数の組を記憶しておく大域変数を  $Q$  とする.

$x^n$  を求めるための具体的な処理は, 以下の通りである.

1.  $Q$  に, 対象頂点と対象変数の組  $(n, x)$  が含まれているか調べる. 含まれている場合は,  $x^n$  を展開せず, そのまま返す. 含まれていない場合は,  $(n, x)$  を  $Q$  に追加する.

```
var a=10;
for (var i=0; i<10; i+=1) {
  a += i;
}
document.writeln(a);
```

図 1 ソースコードの例

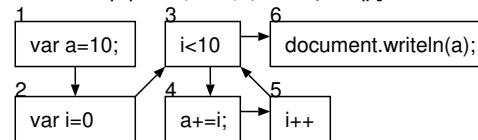


図 2 図 1 のフロー図

2.  $n$  の種類によって, 以下の処理を行う.

対象変数への代入文でない場合  $n$  に入ってくる辺の始点の集合を  $N_{in}$  とし,  $r = \bigcap_{m \in N_{in}} X^m$  とする. 対象変数への代入文で, 右辺が定数の場合  $r = \{ \text{右辺の値} \}$  とする

対象変数への代入文で, 右辺に変数がある場合 右辺に含まれる変数それぞれの値候補を計算する. 右辺の式に, 得られた値候補を代入し,  $r$  に加える.  $r$  の要素に  $x^n$  を含む式があった場合は, その式に  $x^n$  を含まない式を代入し,  $r$  に加えることを,  $r$  の要素が増えなくなるまで繰り返す.

3.  $Q$  から  $(n, x)$  を削除し,  $r$  を返す.

## 3 まとめと今後の課題

本稿では, ECMAScript プログラムをフロー図を用いて静的に解析することによって, 任意の文中における変数の候補を計算するアルゴリズムについて述べた. 今後, 本アルゴリズムを実装して XHTML 解析システムに組み込み, 実際のデータを用いて本アルゴリズムの有効性について実験を行う予定である.

## 参考文献

- [1] ECMA - Standardizing Information and Communication Systems, Standard ECMA-262, ECMAScript Language Specification, 3rd edition <http://www.ecma.ch/ecmal/STAND/ECMA-262.HTM>, December 1999.
- [2] 鷲尾和則, 松下誠, 井上克郎, "JavaScript を含んだ HTML 文書に対するデータフロー解析を用いた構文検証手法の提案", 信学技報, SS2002-22, pp.13-18, 2002.