Micha Reiser

Graduate Candidate	Micha Reiser
Examiner	Prof. Dr. Luc Bläser
Co-Examiner	- -
Subject Area	Software and Systems

Type Inference and Type Checking for JavaScript Strict Mode

Using Hindley Milner Algorithm W Combined with Abstract Interpretation

a)

```
1 "use strict";
2 const logger = {
3     messages: [],
4     log: function (m) {
5         this.messages.push(m);
6     }
7 }
8
9 const log = logger.log;
10 logger.log("Valid");
11 log("Runtime error");
```

b)

```
this-binding.js:5
        this.messages.push(m);
             ^

TypeError: Cannot read property 'messages' of undefined
    at logger.log (this-binding.js:5:7)
    at Object.<anonymous> (this-binding.js:11:1)
    at Module._compile (module.js:541:32)
    at Object.Module._extensions..js (module.js:550:10)
    at Module.load (module.js:458:32)
    at tryModuleLoad (module.js:417:12)
    at Function.Module._load (module.js:409:3)
    at Function.Module.runMain (module.js:575:10)
    at startup (node.js:160:18)
    at node.js:449:3
```

a) Sample Listing b) Executing the Sample Listing Causes a Runtime Error

In recent years, the popularity of JavaScript drastically increased and became a general-purpose language. However, the tool support for program verification is scarce due to the dynamic nature of JavaScript that is hard to be covered using static analysis. Existing verification tools are either limited to simple bug patterns, are based on a super or subset of JavaScript, or only support outdated JavaScript versions.

This work introduces an algorithm for type inference and type checking of JavaScript code written in strict mode. This algorithm combines the Hindley-Milner Algorithm W with abstract interpretation. The type system used is unsound, as the precision of type inference diminishes for reflection-like code that is mainly found in frameworks or libraries.

The defined algorithm has been implemented and is compared to competing type checkers. The evaluation results show that the presented type inference algorithm is precise for a majority of programs. It provides a valuable feedback to programmers if combined with type checking.

```
Initial memory usage 40.06640625
./this-binding.js
Type inference failed for node
  3 |     messages: [],
  4 |     log: function (m) {
> 5 |         this.messages.push(m);
    |              ^
  6 |     }
  7 | }
  8 |
Error: Type inference failure: Potential null pointer
    when accessing  property messages on null or not
     initialized object of type undefined.
```

Output of Static Analyzing the Sample Listing with the Developed Checker