The Patches of the Nopol Automatic Repair System on the Bugs of Defects4J version 1.1.0

Thomas Durieux, Benjamin Danglot, Zhongxing Yu, Matias Martinez, Simon Urli, Martin Monperrus

To cite this version:

Thomas Durieux, Benjamin Danglot, Zhongxing Yu, Matias Martinez, Simon Urli, et al.. The Patches of the Nopol Automatic Repair System on the Bugs of Defects4J version 1.1.0. [Research Report] hal-01480084, Université Lille 1 - Sciences et Technologies. 2017. hal-01480084

HAL Id: hal-01480084
https://hal.archives-ouvertes.fr/hal-01480084
Submitted on 7 Mar 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
The Patches of the Nopol Automatic Repair System on the Bugs of Defects4J version 1.1.0

Thomas Durieux$^1$ Benjamin Danglot$^1$ Zhongxing Yu$^1$
Matias Martinez$^2$ Simon Urli$^1$ Martin Monperrus$^{1,2}$

1) Inria, 2) University of Lille, 3) University of Valenciennes

Technical report hal-01480084
https://hal.archives-ouvertes.fr/hal-01480084

This report lists the test-suite adequate patches found by the Nopol automatic repair system [3] on version 1.1.0 of the Defects4J benchmark of bugs [2]. In total, Nopol finds a test-suite adequate patches for 103 different bugs, as shown in Table 1.

To refer to this document:
@techreport{techrepNopolDefects4j2017,
    TITLE = {{The Patches of the Nopol Automatic Repair System on the Bugs of Defects4J version 1.1.0}},
    AUTHOR = {Durieux, Thomas and Danglot, Benjamin and Zu, Zhongxing and Martinez, Matias and Monperrus, Martin},
    URL = {https://hal.archives-ouvertes.fr/hal-01480084},
    INSTITUTION = {{Université de Lille}},
    YEAR = {2017},
    NUMBER = {hal-01480084},
}

We the Oracle JDK 1.7.0_7 to run Nopol and OpenJDK 1.8.0 for compiling the project Mockito. We give a time budget of three hours to Nopol to find a patch. The experiment has been conducted on machines that are equipped of Xeon X3440 Quad-core processor and 15GB RAM, the machines are part of the Grid5000 infrastructure [1].

Reproduction information:
- Nopol source code: https://github.com/SpoonLabs/nopol/
  Commit used: 47396d2d5c668517bd45be1d878074858eb934f
- Defects4j version: https://github.com/rjust/defects4j/releases/tag/v1.1.0/
- Execution traces: https://github.com/Spirals-Team/defects4j-repair/tree/master/results/2017-march/

1 Chart

Nopol repairs 9 bugs in the project Chart.

1.1 Chart 3

--- /tmp/chart_3_Nopol/source/org/jfree/data/time/TimeSeries.java
+++ /tmp/chart_3_Nopol/source/org/jfree/data/time/TimeSeries.java
@@ -884,3 +884,3 @@
     }
-     if ( removed ) {   
     }

1
Table 1: The number of synthesized test-suite adequate patches for each project.

<table>
<thead>
<tr>
<th>Project</th>
<th># Patches</th>
<th># Bugs</th>
<th>Repair Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chart</td>
<td>9</td>
<td>26</td>
<td>37%</td>
</tr>
<tr>
<td>Closure</td>
<td>56</td>
<td>133</td>
<td>42%</td>
</tr>
<tr>
<td>Lang</td>
<td>4</td>
<td>65</td>
<td>6%</td>
</tr>
<tr>
<td>Math</td>
<td>24</td>
<td>106</td>
<td>22%</td>
</tr>
<tr>
<td>Mockito</td>
<td>2</td>
<td>38</td>
<td>5%</td>
</tr>
<tr>
<td>Time</td>
<td>8</td>
<td>27</td>
<td>29%</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>103</strong></td>
<td><strong>395</strong></td>
<td><strong>26%</strong></td>
</tr>
</tbody>
</table>

Listing 1: "The generated patch for the bug Chart 3"

Number of tests that execute the patch: 86
Execution time: 50 seconds

1.2 Chart 4

--- /tmp/chart_4_Nopol/source/org/jfree/chart/plot/XYPlot.java
+++ /tmp/chart_4_Nopol/source/org/jfree/chart/plot/XYPlot.java
@@ -1606,3 +1606,5 @@
   public void setRenderer (XYItemRenderer renderer) {
     setRenderer(0, renderer);
     + if (renderer!=null) {
     +   setRenderer(0, renderer);
     +   }
   }

Listing 2: "The generated patch for the bug Chart 4"

Number of tests that execute the patch: 18
Execution time: 7 minutes, 32 seconds

1.3 Chart 5

--- /tmp/chart_5_Nopol/source/org/jfree/data/xy/XYSeries.java
+++ /tmp/chart_5_Nopol/source/org/jfree/data/xy/XYSeries.java
@@ -562,3 +562,3 @@
   // append the value to the list...
   - if (this.autoSort) {
   + if (overwritten!=null) {
     this.data.add(-index - 1, new XYDataItem(x, y));

Listing 3: "The generated patch for the bug Chart 5"

Number of tests that execute the patch: 6
Execution time: 38 seconds

1.4 Chart 9

--- /tmp/chart_9_Nopol/source/org/jfree/data/time/TimeSeries.java
+++ /tmp/chart_9_Nopol/source/org/jfree/data/time/TimeSeries.java
@@ -882,3 +882,5 @@
   + if (org.jfree.data.time.TimeSeries.this.data.size() == org.jfree.data.time.
   + TimeSeries.this.range.length()) {
   +   throw new IllegalArgumentException("Requires start <= end.");
   + }

Listing 4: "The generated patch for the bug Chart 9"

Number of tests that execute the patch: 4
Execution time: 34 seconds

1.5 Chart 13
Listing 5: "The generated patch for the bug Chart 13"

Number of tests that execute the patch: 2
Execution time: 42 seconds

1.6 Chart 17

Listing 6: "The generated patch for the bug Chart 17"

Number of tests that execute the patch: 4
Execution time: 33 seconds

1.7 Chart 21

Listing 7: "The generated patch for the bug Chart 21"

Number of tests that execute the patch: 12
Execution time: 37 seconds

1.8 Chart 25
Number of tests that execute the patch: 10
Execution time: 33 seconds

1.9 Chart 26

Listing 8: "The generated patch for the bug Chart 25"

Number of tests that execute the patch: 1
Execution time: 33 seconds

Listing 9: "The generated patch for the bug Chart 26"

Number of tests that execute the patch: 4
Execution time: 33 seconds

2 Closure

Nopol repairs 56 bugs in the project Closure.

2.1 Closure 1

Listing 10: "The generated patch for the bug Closure 1"

Number of tests that execute the patch: 15
Execution time: 2 minutes, 4 seconds

2.2 Closure 2

Listing 11: "The generated patch for the bug Closure 2"

Number of tests that execute the patch: 17
Execution time: 2 minutes, 7 seconds

2.3 Closure 3
Listing 12: "The generated patch for the bug Closure 3"
Number of tests that execute the patch: 10
Execution time: 7 minutes, 8 seconds

2.4 Closure 5

Listing 13: "The generated patch for the bug Closure 5"
Number of tests that execute the patch: 8
Execution time: 1 minute, 51 seconds

2.5 Closure 7

Listing 14: "The generated patch for the bug Closure 7"
Number of tests that execute the patch: 358
Execution time: 3 minutes, 26 seconds

2.6 Closure 8
Listing 15: "The generated patch for the bug Closure 8"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 28 seconds

2.7 Closure 10

Listing 16: "The generated patch for the bug Closure 10"

Number of tests that execute the patch: 8
Execution time: 4 minutes, 28 seconds

2.8 Closure 12

Listing 17: "The generated patch for the bug Closure 12"

Number of tests that execute the patch: 8
Execution time: 7 minutes, 35 seconds

2.9 Closure 14

Listing 18: "The generated patch for the bug Closure 14"

Number of tests that execute the patch: 7
Execution time: 3 minutes, 2 seconds

2.10 Closure 15

--- /tmp/closure_15_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_15_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429,7 +1429,9 @@
    if (diff != null) {
        return "Node tree inequality:" +
            "\n\nTree1:\n" + toStringTree() +
            "\n\nTree2:\n" + tree2.toStringTree() +
            "\n\nSubtree1: " + diff.nodeA.toStringTree() +
            "\n\nSubtree2: " + diff.nodeB.toStringTree();
+        if (com.google.javascript.rhino.Node.this.type < 32) {
+            return "Node tree inequality:" +
+                "\n\nTree1:\n" + toStringTree() +
+                "\n\nTree2:\n" + tree2.toStringTree() +
+                "\n\nSubtree1: " + diff.nodeA.toStringTree() +
+                "\n\nSubtree2: " + diff.nodeB.toStringTree();
    }
}

Listing 19: "The generated patch for the bug Closure 15"

Number of tests that execute the patch: 8
Execution time: 7 minutes, 28 seconds

2.11 Closure 16

--- /tmp/closure_16_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_16_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1445,3 +1445,5 @@
    if (!isEquivalentTo(node2, false, false)) {
        return new NodeMismatch(this, node2);
    } else if (rValue.isOr()) {
+        if (com.google.javascript.jscomp.TypedScopeCreator.DELEGATE_PROXY_SUFFIX.
+            length() == -1 + com.google.javascript.jscomp.TypedScopeCreator.this.
+            functionAnalysisResults.size()) {  
+            return rValue.getJSType();
+        }
    } else if (rValue.isOr()) {
        return new NodeMismatch(this, node2);
    }

Listing 20: "The generated patch for the bug Closure 16"

Number of tests that execute the patch: 12
Execution time: 4 minutes, 12 seconds

2.12 Closure 17

--- /tmp/closure_17_Nopol/src/google/javascript/jscomp/TypedScopeCreator.java
+++ /tmp/closure_17_Nopol/src/google/javascript/jscomp/TypedScopeCreator.java
@@ -1294,3 +1294,5 @@
    // then we can safely use that.
    return rValue.getJSType();
+    if (!com.google.javascript.jscomp.TypedScopeCreator.DELEGATE_PROXY_SUFFIX.
+        length() == -1 + com.google.javascript.jscomp.TypedScopeCreator.this.
+        functionAnalysisResults.size()) {  
+        return rValue.getJSType();
+    }
}

Listing 21: "The generated patch for the bug Closure 17"

Number of tests that execute the patch: 25
Execution time: 2 minutes, 1 second

2.13 Closure 18

--- /tmp/closure_18_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_18_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429,7 +1429,9 @@
    if (diff != null) {
        return "Node tree inequality:" +
            "\n\nTree1:\n" + toStringTree() +
            "\n\nTree2:\n" + tree2.toStringTree() +
            "\n\nSubtree1: " + diff.nodeA.toStringTree() +
            "\n\nSubtree2: " + diff.nodeB.toStringTree();
+            IS_DISPATCHER) {
+            return "Node tree inequality:" +
+                "\n\nTree1:\n" + toStringTree() +
+                "\n\nTree2:\n" + tree2.toStringTree() +
+                "\n\nSubtree1: " + diff.nodeA.toStringTree() +
+                "\n\nSubtree2: " + diff.nodeB.toStringTree();
    }
}

Listing 22: "The generated patch for the bug Closure 18"

Number of tests that execute the patch: 12
Execution time: 1 minute, 1 second
Listing 22: "The generated patch for the bug Closure 18"

Number of tests that execute the patch: 8
Execution time: 1 minute, 51 seconds

2.14 Closure 20

```java
--- /tmp/closure_20_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_20_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429 ,7 +1429 ,9 @@
     if (diff != null) {
         return "Node tree inequality:" +
             "\nTree1:\n" + toStringTree() +
     -   "\nSubtree1: " + diff.nodeA.toStringTree() +
     -   " Subtree2: " + diff.nodeB.toStringTree();
     +   if (com.google/javascript.rhino.Node.this.type < 32) {
     +     return "Node tree inequality:" +
     +         "\nTree1:\n" + toStringTree() +
     +         "\nTree2:\n" + node2.toStringTree() +
     +         "Subtree1: " + diff.nodeA.toStringTree() +
     +         "Subtree2: " + diff.nodeB.toStringTree();
     }
     }
```

Listing 23: "The generated patch for the bug Closure 20"

Number of tests that execute the patch: 8
Execution time: 2 minutes

2.15 Closure 21

```java
--- /tmp/closure_21_Nopol/src/com/google/javascript/jscomp/CheckSideEffects.java
+++ /tmp/closure_21_Nopol/src/com/google/javascript/jscomp/CheckSideEffects.java
@@ -117 ,7 +117 ,9 @@
     if (n == parent.getLastChild()) {
         for (Node an : parent.getAncestors()) {
             int ancestorType = an.getType();
     -     if (ancestorType == Token.COMMA) continue;
     -     if (ancestorType != Token.EXPR_RESULT && ancestorType != Token.BLOCK) return;
     -     else break;
     +     if (com.google.jscomp.CheckSideEffects.PROTECTOR_FN.length() == 0) {
     +         for (Node an : parent.getAncestors()) {
     +             int ancestorType = an.getType();
     +             if (ancestorType == Token.COMMA) continue;
     +             if (ancestorType != Token.EXPR_RESULT && ancestorType != Token.BLOCK) return;
     +             else break;
     +         }
     }
     }
```

Listing 24: "The generated patch for the bug Closure 21"

Number of tests that execute the patch: 2
Execution time: 1 minute, 59 seconds

2.16 Closure 22

```java
--- /tmp/closure_22_Nopol/src/com/google/javascript/jscomp/CheckSideEffects.java
+++ /tmp/closure_22_Nopol/src/com/google/javascript/jscomp/CheckSideEffects.java
@@ -115 ,6 +115 ,8 @@
         if (ancestorType != Token.COMMA) continue;
         if (ancestorType != Token.EXPR_RESULT && ancestorType != Token.BLOCK) return;
         else break;
     }
     }
```

Listing 25: "The generated patch for the bug Closure 22"

Number of tests that execute the patch: 3
2.17 Closure 26

Listing 26: "The generated patch for the bug Closure 26"

Number of tests that execute the patch: 14
Execution time: 3 minutes, 34 seconds

2.18 Closure 28

Listing 27: "The generated patch for the bug Closure 28"

Number of tests that execute the patch: 8
Execution time: 1 minute, 55 seconds

2.19 Closure 29

Listing 28: "The generated patch for the bug Closure 29"

Number of tests that execute the patch: 12
Execution time: 2 minutes
Listing 29: "The generated patch for the bug Closure 30"

Number of tests that execute the patch: 5
Execution time: 1 minute, 45 seconds

2.21 Closure 31

Listing 30: "The generated patch for the bug Closure 31"

Number of tests that execute the patch: 3
Execution time: 1 minute, 54 seconds

2.22 Closure 33

Listing 31: "The generated patch for the bug Closure 33"

Number of tests that execute the patch: 8
Execution time: 1 minute, 52 seconds

2.23 Closure 35
if ((fieldName.length()) != (5)) {
  if (builder.addProperty(fieldName, fieldType, fieldNameNode) == null) {
    // Duplicate field name, warning and skip
    reporter.warning("Duplicate record field " + fieldName,
    sourceName,
    n.getLineNo(), fieldNameNode.getLineNo());
  }
}

Listing 32: "The generated patch for the bug Closure 35"

Number of tests that execute the patch: 26
Execution time: 2 minutes, 43 seconds

2.24 Closure 36

--- /tmp/closure_36_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_36_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1416,7 +1416,9 @@
if (diff != null) {
-  return "Node tree inequality:" +
-  "\nTree1:\n + toStringTree() +
-  "\nTree2:\n + node2.toStringTree() +
-  "\nSubtree1: " + diff.nodeA.toStringTree() +
-  "\nSubtree2: " + diff.nodeB.toStringTree();
+    IS_DISPATCHER) {
+    return "Node tree inequality:" +
+    "\nTree1:\n + toStringTree() +
+    "\nTree2:\n + node2.toStringTree() -
+    "\nSubtree1: " + diff.nodeA.toStringTree() +
+    "\nSubtree2: " + diff.nodeB.toStringTree();
+  }
}

Listing 33: "The generated patch for the bug Closure 36"

Number of tests that execute the patch: 3
Execution time: 1 minute, 48 seconds

2.25 Closure 38

--- /tmp/closure_38_Nopol/src/com/google/javascript/jscomp/CodeConsumer.java
+++ /tmp/closure_38_Nopol/src/com/google/javascript/jscomp/CodeConsumer.java
@@ -244,3 +244,3 @@
boolean negativeZero = isNegativeZero(x);
-  if (x < 0 && prev == '-') {
+  if ((prev == '-')) {
      add(" ");

Listing 34: "The generated patch for the bug Closure 38"

Number of tests that execute the patch: 1665
Execution time: 3 minutes, 11 seconds

2.26 Closure 40

--- /tmp/closure_40_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_40_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1411,7 +1411,9 @@
if (diff != null) {
-  return "Node tree inequality:" +
-  "\nTree1:\n + toStringTree() +
-  "\nTree2:\n + node2.toStringTree() +
-  "\nSubtree1: " + diff.nodeA.toStringTree() +
-  "\nSubtree2: " + diff.nodeB.toStringTree();
+  if (com.google.javascript.rhino.Node.this.type < 32) {
+    return "Node tree inequality:" +
+    "\nTree1:\n + toStringTree() +
+    "\nTree2:\n + node2.toStringTree() +
+    "\nSubtree1: " + diff.nodeA.toStringTree() +
+    "\nSubtree2: " + diff.nodeB.toStringTree();
+  }
}

Listing 35: "The generated patch for the bug Closure 40"
Number of tests that execute the patch: 3
Execution time: 2 minutes, 7 seconds

2.27 Closure 45

Listing 36: "The generated patch for the bug Closure 45"

Number of tests that execute the patch: 3
Execution time: 1 minute, 41 seconds

2.28 Closure 49

Listing 37: "The generated patch for the bug Closure 49"

Number of tests that execute the patch: 68
Execution time: 1 minute, 47 seconds

2.29 Closure 50

Listing 38: "The generated patch for the bug Closure 50"

Number of tests that execute the patch: 4
Execution time: 1 minute, 50 seconds

2.30 Closure 55

12
Listing 39: "The generated patch for the bug Closure 55"

Number of tests that execute the patch: 11
Execution time: 2 minutes, 14 seconds

2.31 Closure 60

Listing 40: "The generated patch for the bug Closure 60"

Number of tests that execute the patch: 31
Execution time: 12 minutes, 12 seconds

2.32 Closure 61

Listing 41: "The generated patch for the bug Closure 61"

Number of tests that execute the patch: 4
Execution time: 1 minute, 39 seconds

2.33 Closure 62
Listing 42: "The generated patch for the bug Closure 62"
Number of tests that execute the patch: 5
Execution time: 1 minute, 32 seconds

2.34 Closure 63

```
--- /tmp/closure_63_Nopol/src/com/google/javascript/jscomp/LightweightMessageFormatter.java
+++ /tmp/closure_63_Nopol/src/com/google/javascript/jscomp/LightweightMessageFormatter.java
@@ -96,4 +96,3 @@
     // at the end of the line
     & & 0 <= charno & & charno < sourceExcerpt.length()) {
         if (error != null) {
```

Listing 43: "The generated patch for the bug Closure 63"
Number of tests that execute the patch: 5
Execution time: 1 minute, 31 seconds

2.35 Closure 66

```
--- /tmp/closure_66_Nopol/src/com/google/javascript/jscomp/TypeCheck.java
+++ /tmp/closure_66_Nopol/src/com/google/javascript/jscomp/TypeCheck.java
@@ -821,3 +821,5 @@
     if (type == null) {
         nullCount ++;
         if (com.google.jscomp.TypeCheck.this.inExterns) {
```

Listing 44: "The generated patch for the bug Closure 66"
Number of tests that execute the patch: 166
Execution time: 2 minutes, 21 seconds

2.36 Closure 67

```
--- /tmp/closure_67_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_67_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1384,7 +1384,9 @@
     if (diff != null) {
         return "Node tree inequality:" +
         "\nTree1:\n" + toStringTree() +
```

Listing 45: "The generated patch for the bug Closure 67"
Number of tests that execute the patch: 2
Execution time: 1 minute, 37 seconds

2.37 Closure 72

```
--- /tmp/closure_72_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_72_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1381,7 +1381,9 @@
     if (diff != null) {
         return "Node tree inequality:" +
         "\nTree1:\n" + toStringTree() +
```

Listing 46: "The generated patch for the bug Closure 72"
Number of tests that execute the patch: 2
Execution time: 1 minute, 32 seconds

Listing 46: "The generated patch for the bug Closure 72"

Number of tests that execute the patch: 3164
Execution time: 24 minutes, 1 second

Listing 47: "The generated patch for the bug Closure 75"

Number of tests that execute the patch: 5
Execution time: 1 minute, 30 seconds

Listing 48: "The generated patch for the bug Closure 76"

Number of tests that execute the patch: 1
Execution time: 1 minute, 25 seconds

Listing 49: "The generated patch for the bug Closure 78"

Number of tests that execute the patch: 1
Execution time: 1 minute, 25 seconds
if (message != null) {
    return;
}

Listing 50: "The generated patch for the bug Closure 111"

Number of tests that execute the patch: 381
Execution time: 2 minutes, 49 seconds

2.42 Closure 113

--- /tmp/closure_113_Nopol/src/com/google/javascript/jscomp/ProcessClosurePrimitives.java
+++ /tmp/closure_113_Nopol/src/com/google/javascript/jscomp/ProcessClosurePrimitives.java
@@ -328,3 +328,3 @@
    // let them be caught in the subsequent run.
    if (provided != null) {
        if ((provided != null) || ((ns.length() < 10))) {

Listing 51: "The generated patch for the bug Closure 113"

Number of tests that execute the patch: 26
Execution time: 2 minutes, 22 seconds

2.43 Closure 114

--- /tmp/closure_114_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_114_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429,7 +1429,9 @@
    if (diff != null) {
        return "Node tree inequality:" +
                "\nTree1:\n" + toStringTree() +
                "\nTree2:\n" + node2.toStringTree() +
                "\nSubtree1: " + diff.nodeA.toStringTree() +
                "\nSubtree2: " + diff.nodeB.toStringTree();
        } else {

Listing 52: "The generated patch for the bug Closure 114"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 14 seconds

2.44 Closure 115

--- /tmp/closure_115_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_115_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429,7 +1429,9 @@
    if (diff != null) {
        return "Node tree inequality:" +
                "\nTree1:\n" + toStringTree() +
                "\nTree2:\n" + node2.toStringTree() +
                "\nSubtree1: " + diff.nodeA.toStringTree() +
                "\nSubtree2: " + diff.nodeB.toStringTree();
        } else {

Listing 53: "The generated patch for the bug Closure 115"

Number of tests that execute the patch: 14
Execution time: 2 minutes, 18 seconds

2.45 Closure 116

--- /tmp/closure_116_Nopol/src/com/google/javascript/rhino/Node.java
+++ /tmp/closure_116_Nopol/src/com/google/javascript/rhino/Node.java
@@ -1429,7 +1429,9 @@
    if (diff != null) {
        return "Node tree inequality:" +
                "\nTree1:\n" + toStringTree() +
                "\nTree2:\n" + node2.toStringTree() +
                "\nSubtree1: " + diff.nodeA.toStringTree() +
                "\nSubtree2: " + diff.nodeB.toStringTree();
        } else {

Listing 54: "The generated patch for the bug Closure 116"

Number of tests that execute the patch: 14
Execution time: 2 minutes, 18 seconds
Listing 54: "The generated patch for the bug Closure 116"

Number of tests that execute the patch: 13
Execution time: 2 minutes, 13 seconds

2.46 Closure 119

Listing 55: "The generated patch for the bug Closure 119"

Number of tests that execute the patch: 394
Execution time: 32 minutes, 2 seconds

2.47 Closure 120

Listing 56: "The generated patch for the bug Closure 120"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 11 seconds

2.48 Closure 121
Listing 57: "The generated patch for the bug Closure 121"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 12 seconds

2.49 Closure 124

Listing 58: "The generated patch for the bug Closure 124"

Number of tests that execute the patch: 8
Execution time: 17 minutes, 49 seconds

2.50 Closure 126

Listing 59: "The generated patch for the bug Closure 126"

Number of tests that execute the patch: 2
Execution time: 2 minutes, 1 second

2.51 Closure 127

Listing 60: "The generated patch for the bug Closure 127"

Number of tests that execute the patch: 13
Execution time: 2 minutes, 4 seconds

2.52 Closure 129
Listing 61: "The generated patch for the bug Closure 129"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 18 seconds

2.53 Closure 130

Listing 62: "The generated patch for the bug Closure 130"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 3 seconds

2.54 Closure 131

Listing 63: "The generated patch for the bug Closure 131"

Number of tests that execute the patch: 9
Execution time: 2 minutes, 21 seconds

2.55 Closure 132
if (diff != null) {
    return "Node tree inequality:" +
    "\nTree1:\n" + toStringTree() +
    "\nTree2:\n" + node2.toStringTree() +
    "\nSubtree1: " + diff.nodeA.toStringTree() +
    "\nSubtree2: " + diff.nodeB.toStringTree();
} else {
    return "Node tree inequality:" +
    "\nTree1:\n" + toStringTree() +
    "\nTree2:\n" + tostringTree() +
    "\nSubtree1: " + diff.nodeA.toStringTree() +
    "\nSubtree2: " + diff.nodeB.toStringTree();
}

Listing 64: "The generated patch for the bug Closure 132"

Number of tests that execute the patch: 8
Execution time: 2 minutes, 25 seconds

2.56 Closure 133

--- /tmp/closure_133_Nopol/src/com/google/javascript/rhino/Node.js
+++ /tmp/closure_133_Nopol/src/com/google/javascript/rhino/Node.js
@@ -1412,4 +1412,3 @@
    if (token == JsDocToken.EOC || token == JsDocToken.EOL ||
        token == JsDocToken.EOF) {
+    if (com.google.javascript.jscomp.parsing.JsDocInfoParser.this.unreadToken != null) {
        return new ExtractionInfo("", token);
    }
}

Listing 65: "The generated patch for the bug Closure 133"

Number of tests that execute the patch: 254
Execution time: 4 minutes, 22 seconds

3. Lang

Nopol repairs 4 bugs in the project Lang.

3.1 Lang 44

--- /tmp/lang_44_Nopol/src/java/org/apache/commons/NumberUtils.java
+++ /tmp/lang_44_Nopol/src/java/org/apache/commons/NumberUtils.java
@@ -192,5 +192,3 @@
    case 'L':
        if (dec == null
            && exp == null
@@ -201,3 +199,3 @@
            try {
                return createBigInteger(numeric);
            }
@@ -201,3 +201,3 @@
    }
    return createBigInteger(numeric);
    
Listing 66: "The generated patch for the bug Lang 44"

Number of tests that execute the patch: 3
Execution time: 29 seconds

3.2 Lang 51

--- /tmp/lang_51_Nopol/src/java/org/apache/commons/BooleanUtils.java
+++ /tmp/lang_51_Nopol/src/java/org/apache/commons/BooleanUtils.java
@@ -676,6 +676,6 @@
    }
    if (ch == 'Y') {
        return
@@ -676,6 +676,6 @@
    }
    if (ch == 'Y') {
        return
    }
    if (str != null) {
+        return
        if (str.charAt(1) == 'E' || str.charAt(1) == 'e') &&
            (str.charAt(2) == 'S' || str.charAt(2) == 's');
+        if (str!=null) {
+            return
+            if (str.charAt(1) == 'E' || str.charAt(1) == 'e') &&

3.3 Lang 53

Listing 68: "The generated patch for the bug Lang 53"

Number of tests that execute the patch: 6
Execution time: 35 seconds

3.4 Lang 58

Listing 69: "The generated patch for the bug Lang 58"

Number of tests that execute the patch: 3
Execution time: 10 minutes, 26 seconds

4 Math

Nopol repairs 24 bugs in the project Math.

4.1 Math 2

Listing 70: "The generated patch for the bug Math 2"

Number of tests that execute the patch: 63
Execution time: 6 minutes, 18 seconds

4.2 Math 4
Listing 71: "The generated patch for the bug Math 4"

Number of tests that execute the patch: 24
Execution time: 6 minutes, 7 seconds

4.3 Math 7

Listing 72: "The generated patch for the bug Math 7"

Number of tests that execute the patch: 14
Execution time: 20 minutes, 6 seconds

4.4 Math 24

Listing 73: "The generated patch for the bug Math 24"

Number of tests that execute the patch: 2
Execution time: 9 minutes, 53 seconds

4.5 Math 28

Listing 74: "The generated patch for the bug Math 28"

Number of tests that execute the patch: 4
Execution time: 10 minutes, 31 seconds

4.6 Math 33
if ( org.apache.commons.math3.optimization.linear.SimplexTableau.this.numSlackVariables < org.apache.commons.math3.optimization.linear.SimplexTableau.this.constraints.size()) {
    columnsToDrop.add(i);
}
}

Listing 75: "The generated patch for the bug Math 33"

Number of tests that execute the patch: 5  
Execution time: 9 minutes, 59 seconds

4.7 Math 39

+++ /tmp/math_39_Nopol/src/main/java/org/apache/commons/math/ode/nonstiff/AdaptiveStepsizeIntegrator.java
@@ -263,3 +263,5 @@
     ratio = yDot0[j] / scale[j];
-    yDotOnScale2 += ratio * ratio;
+this.maxStep)) {
+        yDotOnScale2 += ratio * ratio;
+    }

Listing 76: "The generated patch for the bug Math 39"

Number of tests that execute the patch: 101  
Execution time: 6 minutes, 49 seconds

4.8 Math 40

--- /tmp/math_40_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BracketingNthOrderBrentSolver.java
+++ /tmp/math_40_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BracketingNthOrderBrentSolver.java
@@ -259,3 +259,3 @@
    // we try again with a lower interpolation order
-    if (signChangeIndex - start >= end - signChangeIndex) {
+    if (((signChangeIndex - start) >= (end - signChangeIndex)) || (xA <= org.
    apache.commons.math.analysis.solversBracketingNthOrderBrentSolver.REDUCTION_FACTOR)) {
        // we have more points before the sign change, drop the lowest point

Listing 77: "The generated patch for the bug Math 40"

Number of tests that execute the patch: 19  
Execution time: 10 minutes, 33 seconds

4.9 Math 41

--- /tmp/math_41_Nopol/src/main/java/org/apache/commons/math/stat/descriptive/moment/Mean.java
+++ /tmp/math_41_Nopol/src/main/java/org/apache/commons/math/stat/descriptive/moment/Mean.java
@@ -216,3 +216,5 @@
    )
    return xbarw + (correction/sumw);
+    if (xbarw < length) {
+        return xbarw + (correction/sumw);
+    }

Listing 78: "The generated patch for the bug Math 41"

Number of tests that execute the patch: 6  
Execution time: 4 minutes, 31 seconds

4.10 Math 42

--- /tmp/math_42_Nopol/src/main/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
+++ /tmp/math_42_Nopol/src/main/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
@@ -346,3 +346,5 @@
    if (getBasicRow(col) == null) {
        columnsToDrop.add(col);
+        if (1 <= org.apache.commons.math.optimization.linear.SimplexTableau.this.numSlackVariables) {
+            columnsToDrop.add(col);

23
Listing 79: "The generated patch for the bug Math 42"
Number of tests that execute the patch: 16
Execution time: 4 minutes, 39 seconds

4.11 Math 44

--- /tmp/math_44_Nopol/src/main/java/org/apache/commons/math/ode/AbstractIntegrator.java
+++ /tmp/math_44_Nopol/src/main/java/org/apache/commons/math/ode/AbstractIntegrator.java
@@ -286,3 +286,5 @@
  + if (1 == org.apache.commons.math.ode.AbstractIntegrator.this.stepHandlers.size)
  + statesInitialized = true;
  + }

Listing 80: "The generated patch for the bug Math 44"
Number of tests that execute the patch: 137
Execution time: 7 minutes, 10 seconds

4.12 Math 49

--- /tmp/math_49_Nopol/src/main/java/org/apache/commons/math/linear/OpenMapRealVector.java
+++ /tmp/math_49_Nopol/src/main/java/org/apache/commons/math/linear/OpenMapRealVector.java
@@ -666,3 +666,5 @@
  + if (org.apache.commons.math.linear.OpenMapRealVector.DEFAULT_ZERO_TOLERANCE == org.
  + apache.commons.math.linear.OpenMapRealVector.this.epsilon) {
  + entries.remove(index);
  + }

Listing 81: "The generated patch for the bug Math 49"
Number of tests that execute the patch: 3
Execution time: 3 minutes, 18 seconds

4.13 Math 50

--- /tmp/math_50_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BaseSecantSolver.java
+++ /tmp/math_50_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BaseSecantSolver.java
@@ -236,15 +236,17 @@
  + if ((((atol) != (org.apache.commons.math.analysis.solvers.BaseSecantSolver.DEFAULT_ABSOLUTE_ACCURACY)) && (org.apache.commons.math.analysis.solvers.BaseSecantSolver.this.method != null)) || (f0 <= atol)) {
  + switch (allowed) {
  + case ANY_SIDE:
  + return x1;
  + case LEFT_SIDE:
  + return inverted ? x1 : x0;
  + case RIGHT_SIDE:
  + return inverted ? x0 : x1;
  + case BELOW_SIDE:
  + return (f1 <= 0) ? x1 : x0;
  + case ABOVE_SIDE:
  + return (f1 >= 0) ? x1 : x0;
  + default:
  + throw new MathInternalError();
  + if (((atol) != (org.apache.commons.math.analysis.solvers.BaseSecantSolver.DEFAULT_ABSOLUTE_ACCURACY)) && (org.apache.commons.math.analysis.solvers.BaseSecantSolver.this.method != null)) || (f0 <= atol)) {
  + switch (allowed) {
Listing 82: "The generated patch for the bug Math 50"

Number of tests that execute the patch: 21
Execution time: 8 minutes, 22 seconds

4.14 Math 57

Listing 83: "The generated patch for the bug Math 57"

Number of tests that execute the patch: 5
Execution time: 3 minutes, 10 seconds

4.15 Math 58

Listing 84: "The generated patch for the bug Math 58"

Number of tests that execute the patch: 10
Execution time: 2 minutes, 54 seconds

4.16 Math 69

Listing 85: "The generated patch for the bug Math 69"

Number of tests that execute the patch: 11
Execution time: 33 seconds

4.17 Math 73
if (((1 != (lower))) && (org.apache.commons.math.analysis.solvers.UnivariateRealSolverImpl.this.result < org.apache.commons.math.analysis.solvers.UnivariateRealSolverImpl.this.defaultFunctionValueAccuracy)) || (initial <= lower)) {
    throw MathRuntimeException.createIllegalArgumentException(
        "invalid interval, initial value parameters: lower={0}, initial={1},
        upper={2}",
        lower, initial, upper);
} else {
    throw MathRuntimeException.createIllegalArgumentException(
        "invalid interval, initial value parameters: lower={0}, initial={1}, upper={2}",
        lower, initial, upper);
}

Listing 86: "The generated patch for the bug Math 73"

Number of tests that execute the patch: 5
Execution time: 34 seconds

4.18 Math 78

--- /tmp/math_78_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BrentSolver.java
+++ /tmp/math_78_Nopol/src/main/java/org/apache/commons/math/analysis/solvers/BrentSolver.java
@@ -260 ,3 +260 ,3 @@
        // Linear interpolation.
    if (((1 < x2) && (!(y0 <= i))) || (x0 == x2)) {
        // it should NOT be replaced by proximity test
        if (x0 == x2) {
            if (((1 < x2) && (!(y0 <= i))) || (x0 == x2)) {
                // Linear interpolation.
                if (x0 == x2) {
                    // Linear interpolation.
                    if (x0 == x2) {
                        // Linear interpolation.
                        if (x0 == x2) {
                            // Linear interpolation.
                        }
                    }
                }
            }
        }
    }

Listing 87: "The generated patch for the bug Math 78"

Number of tests that execute the patch: 116
Execution time: 15 minutes, 41 seconds

4.19 Math 80

--- /tmp/math_80_Nopol/src/main/java/org/apache/commons/math/linear/EigenDecompositionImpl.java
+++ /tmp/math_80_Nopol/src/main/java/org/apache/commons/math/linear/EigenDecompositionImpl.java
@@ -1138 ,3 +1138 ,5 @@
        final double tmp = work[i + k];
        if (org.apache.commons.math.linear.EigenDecompositionImpl.this.cachedD != null) {
            work[i + k] = work[j - k];
        }
        work[j - k] = tmp;

Listing 88: "The generated patch for the bug Math 80"

Number of tests that execute the patch: 1
Execution time: 39 seconds

4.20 Math 81

--- /tmp/math_81_Nopol/src/main/java/org/apache/commons/math/linear/EigenDecompositionImpl.java
+++ /tmp/math_81_Nopol/src/main/java/org/apache/commons/math/linear/EigenDecompositionImpl.java
@@ -1585 ,4 +1585 ,4 @@
        case 1 : // one eigenvalue just deflated. use dMin1, dN1 for dMin and dN.
            if (dMin1 == dN1 && dMin2 == dN2) {
                if (org.apache.commons.math.linear.EigenDecompositionImpl.this.main != null) {
                    // cases 7 and 8.
                }
            }
            else {
                // case 9.

Listing 89: "The generated patch for the bug Math 81"

Number of tests that execute the patch: 4
Execution time: 40 seconds

4.21 Math 85

--- /tmp/math_85_Nopol/src/java/org/apache/commons/math/analysis/solvers/UnivariateRealSolverUtils.java
+++ /tmp/math_85_Nopol/src/java/org/apache/commons/math/analysis/solvers/UnivariateRealSolverUtils.java
@@ -197 ,9 +197 ,9 @@

Listing 26
if (fa * fb >= 0.0 ) {
  if (maximumIterations == initial) {
    throw new ConvergenceException(
      "number of iterations={0}, maximum iterations={1}, " +
      "initial={2}, lower bound={3}, upper bound={4}, final a value={5}, " +
      "final b value={6}, f(a)={7}, f(b)={8} ",
      numIterations, maximumIterations, initial,
      lowerBound, upperBound, a, b, fa, fb);
  }
  "number of iterations={0}, maximum iterations={1}, " +
  "initial={2}, lower bound={3}, upper bound={4}, final a value={5}, " +
  "final b value={6}, f(a)={7}, f(b)={8}" ,
  numIterations, maximumIterations, initial,
  lowerBound, upperBound, a, b, fa, fb);
}

Listing 90: "The generated patch for the bug Math 85"

Number of tests that execute the patch: 17
Execution time: 37 seconds

4.22 Math 87

--- /tmp/math_87_Nopol/src/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
+++ /tmp/math_87_Nopol/src/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
@@ -160 ,4 +160 ,6 @@
      getInvertedCoefficientSum (constraint.getCoefficients ());
    }
  }

Listing 91: "The generated patch for the bug Math 87"

Number of tests that execute the patch: 12
Execution time: 50 seconds

4.23 Math 88

--- /tmp/math_88_Nopol/src/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
+++ /tmp/math_88_Nopol/src/java/org/apache/commons/math/optimization/linear/SimplexTableau.java
@@ -338 ,3 +338 ,5 @@
      if ( org.apache.commons.math.optimization.linear.SimplexTableau.this.
        numDecisionVariables) {
        coefficients[i] = 0;
      }
    }

Listing 92: "The generated patch for the bug Math 88"

Number of tests that execute the patch: 3
Execution time: 23 seconds

4.24 Math 105

--- /tmp/math_105_Nopol/src/java/org/apache/commons/math/stat/regression/SimpleRegression.java
+++ /tmp/math_105_Nopol/src/java/org/apache/commons/math/stat/regression/SimpleRegression.java
@@ -107 ,3 +107 ,5 @@
      ybar = y;
    }
  }
}

Listing 93: "The generated patch for the bug Math 105"

Number of tests that execute the patch: 9
Execution time: 35 seconds

5 Mockito

Nopol repairs 2 bugs in the project Mockito.
5.1 Mockito 29

```java
appendQuoting(description);
- description.appendText(wanted.toString());
+ if (org.mockito.internal.matchers.Same.this.wanted != null) {
+     description.appendText(wanted.toString());
+ }
appendQuoting(description);
```

Listing 94: "The generated patch for the bug Mockito 29"

Number of tests that execute the patch: 5
Execution time: 18 seconds

5.2 Mockito 38

```java
public boolean matches(Object actual) {
- if (this.wanted == null) {
+ if (!((actual != null) && (actual != null))) {
     return actual == null;
+ if (((20) - (value + newTypes.length) < value) || ((2) != (org.joda.time.Partial.
+   this.iTypes.length))) {
+     newValues[i] = value;
+ }
System.arraycopy(iTypes, i, newTypes, i + 1, newTypes.length - i - 1);
```

Listing 95: "The generated patch for the bug Mockito 38"

Number of tests that execute the patch: 173
Execution time: 3 minutes, 12 seconds

6 Time

Nopol repairs 8 bugs in the project Time.

6.1 Time 4

```java
newTypes[i] = fieldType;
- newValues[i] = value;
+ if (((20) - (value + newTypes.length) < value) || ((2) != (org.joda.time.Partial.
+   this.iTypes.length))) {
+     newValues[i] = value;
+ }
System.arraycopy(iTypes, i, newTypes, i + 1, newTypes.length - i - 1);
```

Listing 96: "The generated patch for the bug Time 4"

Number of tests that execute the patch: 75
Execution time: 56 seconds

6.2 Time 7

```java
+ saveField(DateTimeFieldType.year(), iDefaultYear);
- return computeMillis(resetFields, text);
+ if (resetFields) {
+     return computeMillis(resetFields, text);
+ }
```

Listing 97: "The generated patch for the bug Time 7"

Number of tests that execute the patch: 84
Execution time: 43 seconds

6.3 Time 11
Listing 98: "The generated patch for the bug Time 11"
Number of tests that execute the patch: 18
Execution time: 53 seconds

6.4 Time 12
Listing 99: "The generated patch for the bug Time 12"
Number of tests that execute the patch: 159
Execution time: 1 minute, 29 seconds

6.5 Time 14
Listing 100: "The generated patch for the bug Time 14"
Number of tests that execute the patch: 63
Execution time: 37 seconds

6.6 Time 16
Listing 101: "The generated patch for the bug Time 16"

Number of tests that execute the patch: 54
Execution time: 40 seconds

6.7 Time 18

--- /tmp/time_18_Nopol/src/main/java/org/joda/time/chrono/BasicChronology.java
+++ /tmp/time_18_Nopol/src/main/java/org/joda/time/chrono/BasicChronology.java
@@ -604,3 +604,5 @@
  FieldUtils.verifyValueBounds(DateTimeFieldType.monthOfYear(), monthOfYear, 1,
  getMaxMonth(year));
- FieldUtils.verifyValueBounds(DateTimeFieldType.dayOfMonth(), dayOfMonth, 1,
  getDaysInYearMonth(year, monthOfYear));
+ if (!(10 == monthOfYear)) && (((31) - (2)) != (dayOfMonth)) {
+   FieldUtils.verifyValueBounds(DateTimeFieldType.dayOfMonth(), dayOfMonth, 1,
+  getDaysInYearMonth(year, monthOfYear));
+}
  return getYearMonthDayMillis(year, monthOfYear, dayOfMonth);

Listing 102: "The generated patch for the bug Time 18"

Number of tests that execute the patch: 3456
Execution time: 7 minutes, 48 seconds

6.8 Time 19

--- /tmp/time_19_Nopol/src/main/java/org/joda/time/DateTimeZone.java
+++ /tmp/time_19_Nopol/src/main/java/org/joda/time/DateTimeZone.java
@@ -899,3 +899,3 @@
- } else if (offsetLocal > 0) {
+ } else if (org.joda.time.DateTimeZone.this.iID != null) {
  long prev = previousTransition(instantAdjusted);

Listing 103: "The generated patch for the bug Time 19"

Number of tests that execute the patch: 2139
Execution time: 3 minutes, 24 seconds

References

