Why does my spreadsheet compute wrong values?

Birgit Hofer, and Franz Wotawa
Graz University of Technology, Austria
Why spreadsheets?

- Used in nearly every company
- Basis for decisions
- Error prone
  - 3-5 % chance to make a fault in a formula
  - 88 % of spreadsheets contain faults
- Hard to debug
  - Size of spreadsheets
  - Structure hidden
Outline

1. Related work
2. Running Example
3. Models
   a. value-based
   b. dependency-based
   c. improved dependency-based
4. Evaluation
5. Conclusion
Spreadsheet Fault Localization

- Spectrum-based Fault Localization
- Model-based Software Debugging
  - Abreu et al. [AHP14]
  - Jannach and Schmitz [JS14]
- Localization by repair


## Running Example

This is a simplified version of the homework/Budgetone spreadsheet from the EUSES Spreadsheet Corpus.

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td>1st Qtr</td>
<td>2nd Qtr</td>
<td></td>
</tr>
<tr>
<td>Units Sold</td>
<td>1000</td>
<td>1500</td>
<td>2500</td>
</tr>
<tr>
<td>ASP/Unit</td>
<td>$20</td>
<td>$21</td>
<td>20,6</td>
</tr>
<tr>
<td>Sales Revenue</td>
<td>$20.000</td>
<td>$31.500</td>
<td>51.500</td>
</tr>
<tr>
<td>Expenses</td>
<td>$5.000</td>
<td>$6.000</td>
<td>5.000</td>
</tr>
<tr>
<td>Operating Income</td>
<td>$15.000</td>
<td>$25.500</td>
<td>46.500</td>
</tr>
<tr>
<td>Op Income in %</td>
<td>75,0 %</td>
<td>81,0 %</td>
<td>90,3 %</td>
</tr>
</tbody>
</table>

The calculation should be 78.6%.
Running Example – Formula View

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Item</strong></td>
<td><strong>1st Qtr</strong></td>
<td><strong>2nd Qtr</strong></td>
</tr>
<tr>
<td>1</td>
<td><strong>Units Sold</strong></td>
<td>1000</td>
<td>1500</td>
</tr>
<tr>
<td>2</td>
<td><strong>ASP/Unit</strong></td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td><strong>Sales Revenue</strong></td>
<td>=B3*B2</td>
<td>=C3*C2</td>
</tr>
<tr>
<td>4</td>
<td><strong>Expenses</strong></td>
<td>5000</td>
<td>6000</td>
</tr>
<tr>
<td>5</td>
<td><strong>Operating Income</strong></td>
<td>=B4-B5</td>
<td>=C4-C5</td>
</tr>
<tr>
<td>6</td>
<td><strong>Op Income in %</strong></td>
<td>=B6/B4</td>
<td>=C6/C4</td>
</tr>
</tbody>
</table>

This is a simplified version of the homework/Budgetone spreadsheet from the EUSES Spreadsheet Corpus.
Running Example – Dependency Graph

This is a simplified version of the homework/Budgetone spreadsheet from the EUSES Spreadsheet Corpus
Model-Based (Software) Debugging

Spreadsheet + Expect. Output → Conversion → Constraints (Model) → Solve → Diagnoses

- **Single Fault:**
  - \{D5\}
  - \{D6\}
  - \{D7\}

- **Double Fault:**
  - \{D3,D4\}
  - …

For single faults:
\[
\text{SUM}(AB(D2), AB(D3), \ldots) = 1
\]

Test case:
- D7 == 0.786
- B7 == 0.750
- …

Should be 78.6%

All models are automatically derived from a faulty spreadsheet and also contain the fault!

Models for a Spreadsheet’s Behavior

**Value-based**
- $D_2 = B_2 + C_2$
- $D_3 = D_4 / D_2$
- $B_4 = B_3 * B_2$

**Dependency-based**
- $\text{ok}(B_2) \land \text{ok}(C_2) \Rightarrow \text{ok}(D_2)$
- $\text{ok}(D_4) \land \text{ok}(D_2) \Rightarrow \text{ok}(D_3)$
- $\text{ok}(B_3) \land \text{ok}(B_2) \Rightarrow \text{ok}(B_4)$

### Spreadsheet Example

<table>
<thead>
<tr>
<th>Item</th>
<th>1st Qtr</th>
<th>2nd Qtr</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Units Sold</td>
<td>1000</td>
<td>1500</td>
<td>=SUM(B2:C2)</td>
</tr>
<tr>
<td>ASP/Unit</td>
<td>20</td>
<td>21</td>
<td>=D4/D2</td>
</tr>
<tr>
<td>Sales Revenue</td>
<td>=B3*B2</td>
<td>=C3*C2</td>
<td>=SUM(B4:C4)</td>
</tr>
<tr>
<td>Expenses</td>
<td>5000</td>
<td>6000</td>
<td>=SUM(B5:B5)</td>
</tr>
<tr>
<td>Operating Income</td>
<td>=B4-B5</td>
<td>=C4-C5</td>
<td>=D4-D5</td>
</tr>
<tr>
<td>Op Income in %</td>
<td>=B6/B4</td>
<td>=C6/C4</td>
<td>=D6/D4</td>
</tr>
</tbody>
</table>
Models for a Spreadsheet’s Behavior

Value-based
- D2 == B2 + C2
- D3 == D4 / D2
- B4 == B3 * B2

+ exact, few diagnoses
- computation time
- Reals: lacking support

Dependency-based
- ok(B2) \land ok(C2) \rightarrow ok(D2)
- ok(D4) \land ok(D2) \rightarrow ok(D3)
- ok(B3) \land ok(B2) \rightarrow ok(B4)

+ fast
+ only Boolean
- many diagnoses


Focus of this work
Improving the Dependency-based Model

- Use $\leftrightarrow$ instead of $\rightarrow$
  - $\text{ok(B2)} \land \text{ok(C2)} \leftrightarrow \text{ok(D2)}$
  - $\text{ok(D4)} \land \text{ok(D2)} \leftrightarrow \text{ok(D3)}$
  - $\text{ok(B3)} \land \text{ok(B2)} \leftrightarrow \text{ok(B4)}$

- Coincidental correctness
  - Conditional like IF-function
  - Abstraction function like MIN, MAX, COUNT
  - Boolean
  - Multiplication by zero
  - Power with 0 or 1 as base number or 0 as exponent

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Item</td>
<td>1st Qtr</td>
<td>2nd Qtr</td>
</tr>
<tr>
<td>Units Sold</td>
<td>1000</td>
<td>1500</td>
<td>$=\text{SUM(B2:C2)}$</td>
</tr>
<tr>
<td>ASP/Unit</td>
<td>20</td>
<td>21</td>
<td>$=\text{D4/D2}$</td>
</tr>
<tr>
<td>Sales Revenue</td>
<td>$=\text{B3*B2}$</td>
<td>$=\text{C3*C2}$</td>
<td>$=\text{SUM(B4:C4)}$</td>
</tr>
<tr>
<td>Expenses</td>
<td>5000</td>
<td>6000</td>
<td>$=\text{SUM(B5:B5)}$</td>
</tr>
<tr>
<td>Operating Income</td>
<td>$=\text{B4-B5}$</td>
<td>$=\text{C4-C5}$</td>
<td>$=\text{D4-D5}$</td>
</tr>
<tr>
<td>Op Income in %</td>
<td>$=\text{B6/B4}$</td>
<td>$=\text{C6/C4}$</td>
<td>$=\text{D6/D4}$</td>
</tr>
</tbody>
</table>
Empirical Evaluation

- Java implementation using
  - Apache POI
  - Minion Constraint solver
- Spreadsheets from Integer corpus
  - Single fault only

94 spreadsheets

63 spreadsheets

→ Timeout (20 minutes) for 31 spreadsheets for Value-based model

31 spreadsheets
Empirical Evaluation

<table>
<thead>
<tr>
<th>Model</th>
<th>63 spreadsheets</th>
<th>31 spreadsheets</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>63 spreadsheets</td>
<td>31 spreadsheets</td>
</tr>
<tr>
<td>Number of single fault diagnoses</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value-based</td>
<td>4.0</td>
<td>-</td>
</tr>
<tr>
<td>Dependency-based</td>
<td>13.2</td>
<td>45.0</td>
</tr>
<tr>
<td>Improved Dep.-based</td>
<td>11.0 (-16.6%)</td>
<td>38.6 (-14.2%)</td>
</tr>
<tr>
<td>Runtime in ms</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Value-based</td>
<td>56,818.8</td>
<td>&gt; 20 minutes</td>
</tr>
<tr>
<td>Dependency-based</td>
<td>32.0</td>
<td>187.4</td>
</tr>
<tr>
<td>Improved Dep.-based</td>
<td>31.6</td>
<td>164.8</td>
</tr>
</tbody>
</table>
Implications

– Still **more diagnoses** than value-based model

+ **Real time** applicable

+ **Arbitrary solver** (only Boolean needed)

+ Debugging of spreadsheets containing **Real numbers**

+ **Correct/wrong** instead of concrete values for cells

+ Approach can be used in **other domains** as well
Summary

Value-based (VB)
D2 = B2 + C2
+ less diagnoses
− high computation time
  1/3: 20 min timeout
  2/3: 1 minute

Dependency-based (DB)
ok(B2) \land ok(C2) \rightarrow ok(D2)
− many diagnoses
  3.3 times more than VB
+ low computation time
  less than 1 second

Improved dep.-based (IDB)
ok(B2) \land ok(C2) \leftrightarrow ok(D2)
~ reduced number of diagnoses
  15% less diagnoses than DB
+ low computation time
  less than 1 second

Coincidental correctness
- Conditional like IF-function
- Abstraction functions
- Boolean
- Multiplication and Power

Thank you! Questions?