Software effort estimation based on Use Cases

Marcio Rodrigo Braz, Silvia Regina Vergilio

2009. 4. 22
Lee, sunkyung

© KAIST SE LAB 2009
Contents

- Introduction
- Background
  - Use Case Point (UCP)
  - Fuzzy set theory
- Suggested method
  - Use case Size Point (USP)
  - Fuzzy Use case Size Point (FUSP)
- Case study
- Conclusion
- Discussion
Introduction (1/2)

- Project estimation based on requirement is readily available and easily assessable\(^1\)
  - But, it’s difficult to select a requirement format has consistent quality
  - Use case can reduce variability in quality and increase format consistency

- **Use Case Point**\(^2\) can be useful for project estimation
  - UCP* is an accepted software size metric\(^3\)  
    - Easy to count
    - Can be used in early phase

3. Software size measurement and productivity rating in a large-scale software development department[ICSE, 1998]
However, the method based on UCP* has some limitations

- UCP* does not consider the internal structure of Use Case
- UCP* has only 3 categories of complexity
  - Complexity categories: simple, average, complex

In this paper,

- Apply new classification of complexity
  - Use Case Size Point(USP)
- Use fuzzy theory for gradual change of complexity
  - Fuzzy Use Case Size Point(FUSP)
Use Case Point

- Derived from Function Point
- Can determine functional size of software in an early stage of a project
Step 1. Weight actors and use cases

Weight actors

- Unadjusted Actor Weight (UAW) = \( \sum (\text{# of actors x weight factor}) \)
  - Simple(1): another system with a defined API
  - Average(2): another system interacting through a protocol
  - Complex(3): a person interacting through a GUI or a web page

Weight use cases

- Unadjusted Use Case Weight (UUCW) = \( \sum (\text{# of use cases x weight factor}) \)
  - Simple(5): 3 or less transactions
  - Average(10): 4 to 7 transactions
  - Complex(15): more than 7 transactions

Calculate Unadjusted Use Case Point (UUCP)

- UUCP = UAW + UUCW
Step 2. Apply adjustment factors

- Apply Technical Complexity Factor (TCF)
  \[ TCF = 0.6 + 0.01 \times \sum_{i=1}^{13} F_i \times W_i \]

- Apply Environmental Factor (EF)
  \[ EF = 1.4 - 0.03 \times \sum_{i=1}^{8} F_i \times W_i \]

- Calculate Use Case Point
  - UCP = Unadjusted UCP x TCF x EF
Step 3. Apply Productivity Factor

- Effort = UCP x Productivity Factor

Productivity Factor (PF)\(^{1,2,3}\)

- Productivity in organization
- Unit: PH per UCP
- The way to obtain
  1. Historical data
  2. Use values presented in the literature
     - FP varies from 20 to 36

3. Applying use cases: a practical guide [Addison-Wesley, 1998]
Fuzzy set theory
- Rigorous mathematical discipline that deals with imprecise, ‘fuzzy’ concepts
- Determine the degree of membership by membership function

<table>
<thead>
<tr>
<th>Degree of pretty</th>
<th>score</th>
<th>weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Little pretty</td>
<td>≤ 5</td>
<td>1</td>
</tr>
<tr>
<td>Pretty</td>
<td>6 to 10</td>
<td>2</td>
</tr>
<tr>
<td>Very pretty</td>
<td>&gt; 10</td>
<td>3</td>
</tr>
</tbody>
</table>

Score: 10
Beauty score = 0.4 * 2 + 0.6 * 3
= 2.6

Score: 8
Beauty score = 2
Use case Size Point (1/5)

[Use Case Point]

- UCP classification
  - Actor
  - Main scenario
  - Alternative scenario

  Apply technical complexity factors

  Apply environmental factors

  Calculate effort

[Use case Size Point]

- UCP classification
  - Actor
  - Main scenario
  - Alternative scenario
  - Precondition
  - Exception
  - Postcondition

  Apply technical complexity factors

  Apply environmental factors

  Calculate effort
Calculating of the unadjusted USP*

**Weight actors (TPA)**
- Total complexity of Actor = \( \sum (\# \text{ of actors} \times \text{weight factor}) \)
  - Simple(2) : 5 or less data
  - Average(4) : 6 to 10 data
  - Complex(6) : more than 10 data

※ Data : provided to or received from the UC

**Weight preconditions (TPPrC)**
- Total complexity of PreCondition = \( \sum \) complexity of precondition
  - Simple(1) : 1 logical expression
  - Average(2) : 2 to 3 logical expressions
  - Complex(3) : more than 3 logical expressions

※ Logical expression : tested by the condition in precondition

* : Use case Size Point
Calculating of the unadjusted USP (cont’d)

- **Weight main and alternative scenarios (PCP, TPCA)**
  - Total complexity of scenario = \( \sum \) complexity of scenario
    - Very simple(4) : 5 or less # of (entities + steps)
    - Simple(6) : 6 to 10 # of (entities + steps)
    - Average(8) : 11 to 15 # of (entities + steps)
    - Complex(12) : 16 to 20 # of (entities + steps)
    - Very complex(16) : more than 20 # of (entities + steps)

- **Weight exceptions (TPE)**
  - Total complexity of Exception = \( \sum \) complexity of exception
    - Simple(1) : 1 logical expression
    - Average(2) : 2 to 3 logical expressions
    - Complex(3) : more than 3 logical expressions
Calculated of the unadjusted USP (cont’d)

- **Weight postconditions (TPPoc)**
  - Total complexity of Postcondition = \( \sum \) complexity of postcondition
    - Simple(1): 3 or less entities
    - Average(2): 4 to 6 entities
    - Complex(3): more than 6 entities
  ※ **Entity: related entities with postcondition**

- **Calculate Unadjusted Use case Size Point (UUSP)**
  - UUSP = TPA + TPPrC + PCP + TPCA + TPE + TPPoc
Apply adjustment factor

- Derived from FP and UCP
  - Each influence value is between 0 and 5 according to its influence in the UC

- Apply Technical Adjustment Factors
  \[ FTA = 0.65 + (0.01 \times \sum_{i=1}^{14} I_i) \]

- Apply Environmental Factors
  \[ FAA = (0.01 \times \sum_{i=1}^{5} I_i) \]

- Calculate the Use case Size Point
  \[ = \text{Unadjusted USP x (FTA} - \text{FAA)} \]
Fuzzy Use case Size Point(1/3)

- Extension of USP for **gradual change** of complexity by fuzzy set theory

- Procedures
  - Fuzzification
    - Discrete classification tables → continuous classification
  - Defuzzification
    - Fuzzy number → real value

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Data</th>
<th>UUSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>≤ 5</td>
<td>2</td>
</tr>
<tr>
<td>Average</td>
<td>6 to 10</td>
<td>4</td>
</tr>
<tr>
<td>Complex</td>
<td>&gt; 10</td>
<td>6</td>
</tr>
</tbody>
</table>
Fuzzy Use Case Size Point (2/3)

Fuzzification

- Trapezoidal number

- $m_i$ = lower value of the linguistic term $T_i$ in the classification table
- $n_i = (m_i + m_{i+1}) / 2$
- $a_i = n_{i-1}$
- $b_i = m_{i+1}$

<table>
<thead>
<tr>
<th>Complexity</th>
<th>Data</th>
<th>UUSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simple</td>
<td>$\leq 5$</td>
<td>2</td>
</tr>
<tr>
<td>Average</td>
<td>6 to 10</td>
<td>4</td>
</tr>
<tr>
<td>Complex</td>
<td>$&gt; 10$</td>
<td>6</td>
</tr>
</tbody>
</table>

Membership degree

© KAIST SE LAB 2009
Defuzzification

- Rule
  - When the value is between $m_i$ and $n_i$
  - When the value is between $n_i$ and $b_i$

\[ dFUSP(x) = (x \cdot USP_i + (x \cdot USP_{i+1} \text{ for simple average complex} ) \]

© KAIST SE LAB 2009
Data set
- 5 modules of one system
  - Real project database of a private company
- Each module corresponding to one month of work

Evaluated metrics
- Function Point
- Use Case Point
- Use case Size Point
- Fuzzy Use case Size Point
Evaluation

- Step 1. Training phase
  - Get the productivity value for each metric
  - Use 1 module

<table>
<thead>
<tr>
<th>Measure</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hour/FP</td>
<td>4.4743</td>
</tr>
<tr>
<td>Hour/UCP</td>
<td>26.0417</td>
</tr>
<tr>
<td>Hour/USP</td>
<td>2.6048</td>
</tr>
<tr>
<td>Hour/FUSP</td>
<td>2.5132</td>
</tr>
</tbody>
</table>
Evaluation (cont’d)

Step 2. Evaluation phase

- Calculate the estimated effort
- Use 4 remaining modules
- Adjustment values
  - FP : 1.1
  - UCP : 0.54
  - USP & FUSP : 1.04
- Estimated effort

\[
\text{effort} = \frac{\text{Module Size (FP, UCP, USP, FUSP)}}{\text{Productivity}}
\]
Case study (4/5)

Result

- Accuracy of result
  - USP and FUSP show better result than UCP
  - But, there’s no great difference between FP and USP/FUSP
  - There’s no great difference between USP and FUSP

<table>
<thead>
<tr>
<th>Module</th>
<th>UCP</th>
<th>FP</th>
<th>USP</th>
<th>FUSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>A2</td>
<td>77.6</td>
<td>60.8</td>
<td>67.9</td>
<td>67.6</td>
</tr>
<tr>
<td>A3</td>
<td>66.5</td>
<td>32.5</td>
<td>25.2</td>
<td>23.7</td>
</tr>
<tr>
<td>A4</td>
<td>39.1</td>
<td>3.4</td>
<td>17</td>
<td>16.1</td>
</tr>
<tr>
<td>A5</td>
<td>65.1</td>
<td>22.1</td>
<td>10.6</td>
<td>11.4</td>
</tr>
</tbody>
</table>
Result (cont’d)

- Analysis of result
  - Internal structure of Use Case can affect the effort
  - If the intervals of categories is close, using fuzzy numbers is not effect
Conclusion

❖ Contribution
  ▪ Showed USP can deduce the size of use cases better than UCP
  ▪ Suggest fuzzy set theory can be considered for gradual classification of complexity

❖ Future work
  ▪ Improvement of FUSP
    • Enlarge the number of elements considered
    • Increasing the difference among the complexity of the category
Discussion

- **Limitation**
  - Use Case is usually not detail
    - It may difficult to count the number of data, entity and logical expression
  - The number of samples is too small
    - Only 1 module is used to get the productivity factor
      - There’s no guarantee the productivity factor is correct
    - Metrics are evaluated by only 4 modules
  - There’s no explanation how they get the adjustment factor
Q & A