Comments on “Property-Based Software Engineering Measurement: Refining the Additivity Properties”

Geert Poels and Guido Dedene, Member, IEEE

Abstract—The recently published measure property set of Briand, Morasca, and Basili [1] establishes the foundation of a real software measurement theory. Unfortunately, a number of inconsistencies related to additivity properties might hinder its acceptance and further elaboration. In this paper, it is shown how to remove the ambiguity in the property definitions.

Index Terms—Software measurement, measurement properties, measurement theory, additivity, module connection strength.

1 INTRODUCTION

One of the top priority areas in software engineering measurement is the definition of valid measures for the internal attributes [2] of software artifacts. Real-time measurements of these attributes, especially those related to complexity, coupling and cohesion, provide developers with continuous feedback on the effectiveness of the design techniques they apply [3]. To ensure the validity of the measures a number of researchers have proposed sets of properties, sometimes also called axioms, that each measure of a measurement concept should satisfy. In this context a measurement concept is a general attribute (e.g., complexity) that can have many different related, sometimes even contradicting, viewpoints (e.g., structural, psychological, computational, logical, lexical/textual and readability complexity [4]). Especially for the measurement of complexity [5], [6], [7], [8], [9], [10] and to a lesser degree coupling [11], a number of property sets have been proposed.

Some of these research efforts result in inconsistent property sets, meaning that the properties cannot simultaneously be fulfilled by the same measure, since different properties require different interpretations of the concept to be measured [12], [13]. For instance, some of Weyuker’s axioms [7] require complexity to be defined as comprehensibility, while for other axioms to be satisfied complexity must be related to size. Even for consistent sets, the axiomatic approach suffers from the flaw that the proposed properties are necessary but not sufficient. As a consequence it is possible (as has been demonstrated in [14]) to define measures that satisfy all axioms while at the same time being completely meaningless and not related to any intuitive notion regarding the measured concept.

To make things even worse, conflicting properties may occur in the sets of different authors. As a consequence, a measure may be valid according to one property set, but invalid according to another set.

Consider for instance the complexity properties related to additivity. The relevant question is whether the complexity of a system composed of two software entities (e.g., program segments, modules, object classes, ...) is equal to the sum of the complexities of the individual entities. In [5], the complexity of a system is greater than or equal to the sum of the complexities of its parts. Also in [7], it is stated that complexity does not have to be additive. In fact, the complexity of the whole may be strictly greater than the summed complexities of the parts. In [8], strict additivity is required when program segments are sequenced. Other authors require additivity for some, but not all, viewpoints on complexity [6], [9], [10].

The recently published property set of Briand, Morasca, and Basili is a clear exception in this pattern [1]. The range of concepts covered by the properties is wide: size, length, complexity, coupling, and cohesion. The properties may be used to classify measures according to the concept they are purported to measure. Secondly, in [1] it is nowhere claimed that this set is sufficient to validate a measure. At most the properties are necessary, which implies they can be used to invalidate measures. Further it is argued that the property set is generic. Software systems are represented in an abstract model using a graph-theoretic approach, allowing to be mapped to a wide range of software artifacts (e.g., USES graphs, Is-Component-of graphs, control flow graphs, etc. [1]). Moreover, software systems and the properties that characterise the measurement concepts are formally described using set-theoretic notation, which should guarantee precise and unambiguous definitions. Also, it is shown that the properties do not contradict most axioms formulated in previous axiom sets, because they are weak enough to capture the corresponding parts in otherwise conflicting properties. Finally, Briand, Morasca, and Basili explicitly relate their properties to existing concepts of measurement theory (see e.g., [15]), showing the similarities in both approaches.

It is hoped that this property set is widely discussed and accepted in the software measurement community, both by scientists and practitioners. Its wide coverage of measurement concepts, the formality of its definitions, and its correspondence with earlier published axiom sets and measurement theory are clear advantages not exhibited by previous research. Besides, the explicit recognition that the properties alone are not sufficient to validate a measure is an invitation for continuous improvement. Briand, Morasca, and Basili invite other researchers to extend their set to other concepts (e.g., reuse), to formulate additional properties or to refine existing properties. The convergence of all these ideas could ultimately lead to the foundation of a real software measurement theory.

Unfortunately, the mathematical treatment in [1] is just not precise enough to guarantee an unambiguous interpretation of the property definitions. Especially for the additivity-related properties there exist a number of inconsistencies that might result in confusion and hinder the further elaboration of the framework. The inconsistencies in [1] can be traced back to the definition of a union operator for software entities, that, while being sufficiently formal and correct, is not conscientiously applied in the rest of the text.

We believe that to reach a consensus on a universal software measurement theory, the relationship between the measurement concepts and their additivity properties must be further clarified. The ambiguity in the definition of the properties introduced in [1] can possibly result in wasted research time and costs if it is not recognised and removed. In Section 2, the basic definitions of the software system and the additivity-related properties, such as presented in [1], are reproduced. In Section 3, the inconsistencies in the definitions are removed by introducing an ordinal scale of connection strength between software modules, and by relating this scale to the additivity properties. In Section 4, the origin of the inconsistencies is traced back to the definition of a module union operator, and an alternative definition is proposed. Finally, in Section 5 conclusions are presented.
2 BASIC DEFINITIONS

All definitions presented in this section are taken from [1].

DEFINITION 1. A system $S$ is a pair $<E, R>$, where $E$ is the set of elements of $S$, and $R \subseteq E \times E$ is a binary relation on $E$ representing the relationships between the elements of $S$.

The representation of a system in [1] is general enough to be mapped to a wide range of software artifacts. For instance, if $S$ is a control flow graph of a program, then $E$ is the set of code statements (i.e., the nodes in the graph) and $R$ is the set of control flows between code statements (i.e., the edges in the graph).

DEFINITION 2. Given a system $S = <E, R>$ and a system $m = <Em, Rm>$:

- $m$ is a module of $S$ if $m \subseteq S \Rightarrow Em \subseteq E$ and $Rm \subseteq R$.

Modules are systems in their own right that are contained in other systems. For instance, a module may be the control flow graph of a subprogram. Modules may be connected with the rest of the system through relationships. For a module $m$, the incoming and outgoing relationships are captured by the sets $\text{Input}(m)$ and $\text{Output}(m)$, respectively.

DEFINITION 3. Given a system $S = <E, R>$ and a system $m = <Em, Rm>$ such that $m$ is a module of $S$:

- $\text{Input}(m) = \{e_1, e_2 \in E \mid e_2 \in Em \text{ and } e_1 \in E - Em\}$;
- $\text{Output}(m) = \{e_1, e_2 \in E \mid e_1 \in Em \text{ and } e_2 \in E - Em\}$.

To define meaningful measurement concept properties a number of operators are defined on modules.

DEFINITION 4. Given a system $S = <E, R>$, a system $m_i = <Em_i, Rm_i>$ and a system $m_j = <Em_j, Rm_j>$ where $m_i$ and $m_j$ are modules of $S$:

1) the union of $m_i$ and $m_j$ (notation: $m_i \cup m_j$) is the system $<Em_i \cup Em_j, Rm_i \cup Rm_j>$ which is a module of $S$;
2) the intersection of $m_i$ and $m_j$ (notation: $m_i \cap m_j$) is the system $<Em_i \cap Em_j, Rm_i \cap Rm_j>$ which is a module of $S$;
3) the modules $m_i$ and $m_j$ are disjoint if $Em_i \cap Em_j = \emptyset$;
4) the modules $m_i$ and $m_j$ are not connected $\Rightarrow Em_i \cap Em_j = \emptyset$ and $\text{Input}(m_i) \cap \text{Output}(m_j) = \emptyset$ and $\text{Input}(m_j) \cap \text{Output}(m_i) = \emptyset$.

These concepts are illustrated using the system $S = <E, R>$ of Fig. 1.

![Fig. 1. The system S = <E, R> [1].](image)

The system $S = <E, R>$ is fully described by $<\{a, b, c, d, e, f, g, h, i, j, k, l, m\}, \{<b, a>, <b, f>, <b, d>, <c, d>, <c, g>, <d, f>, <e, g>, <f, k>, <f, i>, <f, k>, <g, m>, <h, a>, <h, j>, <i, j>, <k, j>, <k, l>, <k, b>\}>$.

Four modules of $S$ are identified in the figure:

- $m_1 = <\{a, b, f, i, j, k\}, \{<b, a>, <b, f>, <f, i>, <f, j>, <k, f>, <k, j>, <k, l>, <k, b>\}>$
- $m_2 = <\{c, d, e, f, g, j, k, m\}, \{<c, d>, <c, g>, <d, f>, <e, g>, <f, k>, <g, m>, <k, j>, <k, l>, <k, b>\}>$
- $m_3 = <\{b, a>, <b, f>, <f, i>, <f, k>, <g, m>, <h, a>, <h, j>, <i, j>, <k, j>, <k, l>, <k, b>\}>$
- $m_4 = <\{d, e, g\}, \{<e, g>, <f, k>\}>$

1. In [1], modules $m_i$ and $m_j$ are said to be disjoint if $m_i \cap m_j = \emptyset$. But since $Em_i \cap Em_j = \emptyset \Rightarrow Rm_i \cap Rm_j = \emptyset$, Definition 4.3 is equivalent to the original definition.

Notation $m_1 \subseteq S, m_2 \subseteq S, m_3 \subseteq S, m_4 \subseteq S$ and $m_2 \subseteq m_1 \cap m_3$.

Note that $m_2$ is the intersection of $m_1$ and $m_3$ (i.e., $m_2 = m_1 \cap m_3$). The union of $m_1$ and $m_3$ is given by $m_1 \cup m_3 = \{<a, b, c, d, e, f, g, i, j, k, m>, <b, a>, <b, f>, <c, d>, <c, g>, <d, f>, <e, g>, <f, k>, <f, i>, <f, k>, <g, m>, <h, a>, <h, j>, <i, j>, <k, j>, <k, l>, <k, b>\}$.

Note also that $e \in E_{m_1}$ and $<c, b> \in R$, but $<c, b> \notin R_{m_2}$.

According to [1] the coupling and cohesion concepts are only meaningful in the context of modular systems. A system is a modular system if its set of elements is partitioned into the sets of elements of its modules. As a consequence, all modules in a modular system are disjoint.

DEFINITION 5. A modular system $MS = <E, R, M>$ is a system $<E, R>$, where $M$ is a set of modules such that:

1) $\forall e \in E, \exists m = <Em, Rm> \in M$ such that $m \subseteq S$ and $e \in E_m$;
2) $\forall m_i = <Em_i, Rm_i>, m_j = <Em_j, Rm_j> \in M: m_i$ and $m_j$ are disjoint.

Relationships in a modular system are either intermodule or intramodule.

DEFINITION 6. Given a modular system $MS = <E, R, M>$, where $M = \{m_1, m_2, ..., m_n\}$ and $m_i = <Em_i, Rm_i>$ for $i = 1, ..., n$: the set of intramodule relationships is:

$$IR = \bigcup_{i=1}^{n} R_{m_i}$$

the set of intermodule relationships is $R - IR$.

These concepts are illustrated using the modular system $MS = <E, R, M>$ of Fig. 2.

![Fig. 2. The system MS = <E, R, M> [1].](image)

$MS = <E, R, \{m_1, m_2, m_3\}>$

$IR = \{<b, a>, <c, d>, <c, g>, <e, g>, <f, i>, <f, k>, <g, m>, <h, a>, <h, i>, <i, j>, <k, j>, <k, l>, <k, b>\}$

$R - IR = \{<b, f>, <c, b>, <d, f>, <h, i>, <i, j>\}$

According to Briand et al., Size, Complexity, and Coupling measures should be additive when systems are made of disjoint modules [1]. Hence, for each of these measurement concepts a disjoint module additivity property is defined.

- **Size.** Given a system $S = <E, R>$, a system $m_i = <Em_i, Rm_i>$ and a system $m_j = <Em_j, Rm_j>$:

  $$(m_1 \subseteq S \text{ and } m_2 \subseteq S \text{ and } E = E_{m_1} \cup E_{m_2} \text{ and } E_{m_1} \cap E_{m_2} = \emptyset) \Rightarrow \text{Size}(S) = \text{Size}(m_1) + \text{Size}(m_2)$$

- **Complexity.** Given a system $S = <E, R>$, a system $m_i = <Em_i, Rm_i>$ and a system $m_j = <Em_j, Rm_j>$:

  $$(S = m_1 \cup m_2 \text{ and } m_1 \cap m_2 = \emptyset) \Rightarrow \text{Complexity}(S) = \text{Complexity}(m_1) + \text{Complexity}(m_2).$$

2. In [1], the relationship $<c, b>$ is included in $m_1 \cup m_2$. This is not correct, according to the definition of the module union operator.
• Coupling. Given a modular system MS' = <E, R, M'> and a modular system MS'' = <E, R, M''>, with the same underlying system <E, R>, such that M'' = M' – {m', m''} ∪ {m''}, with m', m'' ∈ M', m'' = m' ∪ m' and m'' ∈ M' and InputR(m') ∩ OutputR(m'') = ∅ and InputR(m') ∩ OutputR(m'') = ∅, then:
  1) Coupling(m'') = Coupling(m') + Coupling(m'').
  2) Coupling(MS') = Coupling(MS'').

According to the same authors, Length and Cohesion measures are not additive. Length measures for systems made up of disjoint modules must satisfy the following nonadditive property:

• Length. Given a system S = <E, R>, a system m_i = <E_m_i, R_m_i> and a system m_j = <E_m_j, R_m_j>; S = m_i ∪ m_j and m_i ∩ m_j = ∅ and E = E_m_i ∪ E_m_j) ⇒ Length(S) = max(Length(m), Length(m_j)).

Cohesion measures satisfy the following nonadditive property:

• Cohesion. Given a modular system MS' = <E, R, M'> and a modular system MS'' = <E, R, M''>, with the same underlying system <E, R>, such that M'' = M' – {m', m''} ∪ {m''}, with m', m'' ∈ M', m'' = m' ∪ m' and m'' ∈ M' and InputR(m') ∩ OutputR(m'') = ∅ and InputR(m') ∩ OutputR(m'') = ∅, then:
  1) max(Cohesion(m'), Cohesion(m'')) ≥ Cohesion(m'')
  2) Cohesion(MS') ≥ Cohesion(MS'').

### 3 Additivity and Connection Strength

The modules in a software system can be connected with each other in many possible ways. Some modules may have common relationships, while other modules may only have common elements. Disjoint modules can be connected by intermodule relationships or can be unconnected. The connection strength between modules determines the nature of the additivity-related properties, i.e., certain measurement concepts are additive given some assumptions concerning the connection strength of the modules. If these assumptions are not met, the additivity property does not hold. To clarify the relation between additivity and connection strength, first an ordinal scale of connection strength is defined. Next, this scale will be used to examine the connection-related properties in [1].

DEFINITION 7. Given a system S = <E, R>, a system m_i = <E_m_i, R_m_i> and a system m_j = <E_m_j, R_m_j>, with m_i, m_j ⊆ S. The conditions for the different levels of connection strength are:

- **Level 0.** No restrictions on connection types
  - Level 1. InputR(m_i) ∩ OutputR(m_j) = ∅ and InputR(m_i) ∩ OutputR(m_j) = ∅.
  - Level 2. R_m_i ∩ R_m_j = ∅.
  - Level 3. R_m_i ∩ R_m_j = ∅ and InputR(m_i) ∩ OutputR(m_j) = ∅ and InputR(m_i) ∩ OutputR(m_j) = ∅.
  - Level 4. E_m_i ∩ E_m_j = ∅.
  - Level 5. E_m_i ∩ E_m_j = ∅ and InputR(m_i) ∩ OutputR(m_j) = ∅ and InputR(m_i) ∩ OutputR(m_j) = ∅.

According to Definition 4, level 4 modules are disjoint and Level 5 modules are not connected. Since the condition for level 4 modules is contained in the condition for level 5 modules, not connected modules are always disjoint. Note however, that the opposite is not true: disjoint modules may be connected. Modular systems are by definition composed of disjoint modules. Hence, the connection strength in a modular system is level 4 or level 5.

Since for some levels the condition is contained in the condition of another level, an ordinal scale [15] of connection strength can be defined.

DEFINITION 8.

1. For the system S of Fig. 1 the following statements hold:
   1) At level 4, modules are disjoint ⇒ m_1 and m_4 are disjoint; m_2 and m_3 are disjoint.
   2) At level 5, modules are not connected ⇒ All modules are connected to some degree.

   The connection strengths between the modules of the system S = <E, R> of Fig. 1 and the modular system MS = <E, R, M> of Fig. 2 are tabulated in Tables 1 and 2, respectively.

#### TABLE 1
**Connection Strength Between Modules in**

<table>
<thead>
<tr>
<th>Connection strength</th>
<th>Module m_1</th>
<th>Module m_2</th>
<th>Module m_3</th>
<th>Module m_4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>module m_1</td>
<td>level 1</td>
<td>level 0</td>
<td>level 4</td>
</tr>
<tr>
<td></td>
<td>module m_2</td>
<td>level 1</td>
<td>level 1</td>
<td>level 4</td>
</tr>
<tr>
<td></td>
<td>module m_3</td>
<td>level 0</td>
<td>level 1</td>
<td>level 1</td>
</tr>
<tr>
<td></td>
<td>module m_4</td>
<td>level 4</td>
<td>level 4</td>
<td>level 4</td>
</tr>
</tbody>
</table>

For the system S of Fig. 1 the following statements hold:

1. At level 4, modules are disjoint ⇒ m_1 and m_4 are disjoint; m_2 and m_3 are disjoint.
2. At level 5, modules are not connected ⇒ All modules are connected to some degree.

In the modular system MS of Fig. 2 each module is connected to each of the other modules.

The inconsistencies introduced in [1] are related to both the concepts of additivity and connection strength. More precisely, according to Briand, Morasca, and Basili for some measurement concepts like complexity and coupling additivity holds when modules are disjoint (i.e., connection strength level 4). Consequently, these properties are called disjoint module additivity properties. However, based on the definitions of these properties provided in [1] for most concepts additivity holds only when modules are not connected (i.e., connection strength level 5). Since disjoint modules may be connected (i.e., <5, 5> <5, 5>), the definition of the properties contradicts the assertion that complexity and coupling are additive for (all) disjoint modules. To formally show this, the following theorem is needed (a proof can be found in the Appendix).

THEOREM 1. Given a system S = <E, R>, a system m_i = <E_m_i, R_m_i> and a system m_j = <E_m_j, R_m_j>:

3. A function µ: level i → {0, 1, 2, 3, 4, 5}: µ(level 0) = 3, µ(level 1) = 2, µ(level 2) = 2, µ(level 3) = 1, µ(level 4) = 1, µ(level 5) = 0 is a valid ordinal measure of the connection strength between modules since it satisfies ∀ levels i, j ∈ {0, 1, 2, 3, 4, 5}: i < j ⇒ µ(level i) < µ(level j).
Given a system $S = <E, R>$, a system $m_i = <E_{mi}, R_{mi}>$ and a system $m_j = <E_{mj}, R_{mj}>$

$(S = m_i \cup m_j$ and $m_i \cap m_j = \emptyset$ and $E = E_{mi} \cup E_{mj}$ ) $\Rightarrow$ $\text{Length}(S) = \max(\text{Length}(m_i), \text{Length}(m_j))$

$E = E_{mi} \cup E_{mj}$ is already implied by $S = m_i \cup m_j$

$\Rightarrow$ $\text{InputR}(m_i) \cap \text{OutputR}(m_i) = \emptyset$ and $\text{InputR}(m_i) \cap \text{OutputR}(m_i) = \emptyset$

(Theorem 1)

$\Rightarrow$ $m_i$ and $m_j$ are not connected (level 5).

Since disjoint modules (level 4) may be connected they do not satisfy this nonadditivity property.

### 3.4 Size

Given a system $S = <E, R>$, a system $m_i = <E_{mi}, R_{mi}>$ and a system $m_j = <E_{mj}, R_{mj}>$

$(m_i \subseteq S$ and $m_j \subseteq S$ and $E = E_{mi} \cup E_{mj}$ and $E_{mi} \cap E_{mj} = \emptyset) \Rightarrow \text{Size}(S) = \text{Size}(m_i) + \text{Size}(m_j)$

Disjoint modules (level 4) satisfy this size additivity property, regardless whether they are connected or not. Since $5 < 4$, the property also applies for not connected components (i.e., not connected components are disjoint by definition). Besides, size is only related to the elements of a system and not to relationships. Therefore, only connections through common elements matter. Disjoint modules do not have common elements.

### 3.5 Cohesion

Given a modular system $MS^* = <E, R, M^*>$ and a modular system $MS" = <E, R, M">$, with the same underlying system $<E, R>$, such that $M^" = M^* - \{m'_i, m'_j\} \cup \{m"\}$, with $m'_i \in M', m" = m'_j \cup m'_i$ and $m" \notin M$ and $\text{InputR}(m'_i) \cap \text{OutputR}(m'_i) = \emptyset$ and $\text{InputR}(m'') \cap \text{OutputR}(m'') = \emptyset$

$$\text{max}(\text{Cohesion}(m'_i), \text{Cohesion}(m'')) \geq \text{Cohesion}(m")$$

$$\text{Cohesion}(MS^*) \geq \text{Cohesion}(MS")$$

Analogous to the definition of the coupling additivity property, modules may not be connected (level 5). Curiously, and opposed to the definition of the coupling additivity property, in [1] it is not claimed that this property holds for all modules in a modular system.

These results are summarized in Table 3. In each cell the level(s) of connection strength are indicated. Note that levels 0 to 3 are not defined for modular systems such as required for the measurement of cohesion and coupling.

### Table 3

<table>
<thead>
<tr>
<th>Concept/Property</th>
<th>additive relation</th>
<th>nonadditive relation</th>
<th>no definite relation</th>
</tr>
</thead>
<tbody>
<tr>
<td>size</td>
<td>4-5</td>
<td>0, 1, 2, 3</td>
<td></td>
</tr>
<tr>
<td>length</td>
<td>5</td>
<td>0, 1, 2, 3, 4</td>
<td></td>
</tr>
<tr>
<td>complexity</td>
<td>5</td>
<td>0, 1, 2, 3, 4</td>
<td></td>
</tr>
<tr>
<td>cohesion</td>
<td>5</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>coupling</td>
<td>5</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

In [1] a similar table (Table 1, p. 81) appears in which it is claimed that complexity and coupling are additive for both level 4 and level 5 modules. We believe this table must be refined to accommodate the notion of connection strength between modules.

### 4 An Alternative Definition of the Union-Operator

The origin of the inconsistencies in [1] can be traced back to confusion arisen by the definition of the union-operator of modules. Given the union-operator such as defined in [1] Theorem 2 (a proof can be found in the Appendix) states that a system is equal
to the union of its modules if and only if these modules are not connected through common incoming and outgoing relationships.

THEOREM 2. Given a system \( S = \langle E, R \rangle \), a system \( m_i = \langle E_{m_i}, R_{m_i} \rangle \) and a system \( m_j = \langle E_{m_j}, R_{m_j} \rangle \) such that \( m_i \) and \( m_j \) are modules of \( S \) and \( E = E_{m_i} \cup E_{m_j} \),

\[
S = m_i \cup m_j \iff \text{InputR}(m_i) \cap \text{OutputR}(m_j) = \emptyset \text{ and } \text{InputR}(m_j) \cap \text{OutputR}(m_i) = \emptyset.
\]

Whenever an incoming relationship of one module is an outgoing relationship of the other module, the system is no longer the union of its modules. An alternative definition of the union operator can be given such that whenever \( E = E_{m_i} \cup E_{m_j} \) it holds that \( S = \langle E, R \rangle \) is the union of its modules \( m_i = \langle E_{m_i}, R_{m_i} \rangle \) and \( m_j = \langle E_{m_j}, R_{m_j} \rangle \) (Theorem 3; a proof can be found in the Appendix).

DEFINITION 9. The union of \( m_i \) and \( m_j \) (notation \( m_i \cup m_j \)) is the system \( \langle E_{m_i} \cup E_{m_j}, R_{m_i} \cup R_{m_j} \rangle \) such that \( m_i \) and \( m_j \) are modules of \( S \) and \( E = E_{m_i} \cup E_{m_j} \).

THEOREM 3. Given a system \( S = \langle E, R \rangle \), a system \( m_i = \langle E_{m_i}, R_{m_i} \rangle \) and a system \( m_j = \langle E_{m_j}, R_{m_j} \rangle \) such that \( m_i \) and \( m_j \) are modules of \( S \) and \( E = E_{m_i} \cup E_{m_j} \),

\[
S = m_i \cup m_j
\]

The properties of \( \cup \) are: Given a system \( m_i = \langle E_{m_i}, R_{m_i} \rangle \) and a system \( m_j = \langle E_{m_j}, R_{m_j} \rangle \) such that \( E_{m_i} \cap E_{m_j} = \emptyset \):

1) \((m_i \cup m_j) \subseteq (m_i \cup m_j)\)
2) If \( m_i \) and \( m_j \) are not connected, then \((m_i \cup m_j) = (m_i \cup m_j)\)
3) If \( S = \langle E, R \rangle = m_i \cup m_j \), then \( m_i \) and \( m_j \) are not connected
4) If \( S = \langle E, R \rangle = m_i \cup m_j \), then \( m_i \) and \( m_j \) are connected and only if \((m_i \cup m_j) = (m_i \cup m_j)\).

We believe that while having defined the union-operator as \( \cup \) (cf. Definition 4), Briand, Morasca, and Basili use it as if it were the union-operator \( \cup \) (cf. Definition 9).

For the coupling disjoint module additivity property it is explicitly required that no intermodule relationships between the modules exist. Therefore, for all modules \( m_i \) and \( m_j \) that satisfy the conditions of the property, \( m'' = m_i \cup m_j \) is the union of \( m_i \) and \( m_j \), since \( m_i \) and \( m_j \) may not be connected. Therefore, in this case we believe ambiguity can be removed by changing the name of the property from disjoint module additivity to not connected module additivity. At least for length and complexity the relationship between additivity-related properties and connection strength must be formally defined since ambiguities (e.g., \( \cup \) or \( \cup \)) may lead to measures being accepted or rejected for the wrong reasons.

5 CONCLUSION

Briand, Morasca, and Basili established the foundation of a formal and rigorous approach to property-based software engineering measurement. Compared to property sets that were previously published, their set shows a number of benefits. The wide coverage of measurement concepts, the formality of the definitions, and the correspondence with earlier published axiom sets and meas-

\[
\begin{align*}
\text{Length}(m_i) &= 3 \\
\text{Length}(m_j) &= 1 \\
\text{Length}(S_{14}) &= 3 \\
\text{Length}(S_{14}') &= 4 \\
\end{align*}
\]

While Length satisfies the nonadditivity property defined with the union operator \( \cup \) (as in [1]), it does not satisfy the nonadditivity property defined with the alternative operator \( \cup \).

4.3 Complexity

Given a system \( S = \langle E, R \rangle \), a system \( m_i = \langle E_{m_i}, R_{m_i} \rangle \) and a system \( m_j = \langle E_{m_j}, R_{m_j} \rangle \) such that \( S = m_i \cup m_j \) and \( m_i \cap m_j = \emptyset \) \Rightarrow Complexity(S) = Complexity(m_i) + Complexity(m_j). Again, only connection strength level 4 (disjoint modules) is required.

Suppose a measure Complexity(S) is defined as the cardinality of \( R \), where \( S = \langle E, R \rangle \). Calculating Complexity for \( m_i, m_j \), \( S_{14} = m_1 \cup m_4 \) and \( S_{14}' = m_1 \cup m_4 \) results in:

- Complexity\( (m_i) \) = 6
- Complexity\( (m_j) \) = 2
- Complexity\( (S_{14}) \) = 8
- Complexity\( (S_{14}') \) = 9

Complexity satisfies additivity if union is defined with \( \cup \), but not if union is defined with \( \cup \).

4.4 Cohesion

No difference since only intramodule relationships are considered in the concept properties. Hence, it does not matter whether the union of modules contains the intermodule relationships between the modules or not.

4.5 Coupling

For the coupling disjoint module additivity property it is explicitly required that no intermodule relationships between the modules exist. Therefore, for all modules \( m_i' \) and \( m_j' \) that satisfy the conditions of the property, \( m'' = m_i' \cup m_j' \) is the union of \( m_i' \) and \( m_j' \), since \( m_i' \) and \( m_j' \) may not be connected. Therefore, in this case we believe ambiguity can be removed by changing the name of the property from disjoint module additivity to not connected module additivity. At least for length and complexity the relationship between additivity-related properties and connection strength must be formally defined since ambiguities (e.g., \( \cup \) or \( \cup \)) may lead to measures being accepted or rejected for the wrong reasons.

\[
\begin{align*}
\text{Length}(m_i) &= 3 \\
\text{Length}(m_j) &= 1 \\
\text{Length}(S_{14}) &= 3 \\
\text{Length}(S_{14}') &= 4 \\
\end{align*}
\]

While Length satisfies the nonadditivity property defined with the union operator \( \cup \) (as in [1]), it does not satisfy the nonadditivity property defined with the alternative operator \( \cup \).

Fig. 3. Adding \( e_d \) to the system S.
urement theory are all definite advantages not exhibited by previous research. However, to reach a consensus on a universal software measurement theory, a number of inconsistencies that may result in ambiguity and confusion must be removed. These inconsistencies pertain to the properties that are related to additivity. They are a result of an improper use of the union-operator for system modules.

In this paper, it has been shown that by introducing the concept of connection strength between system modules, properties related to additivity can be reformulated such that all ambiguity is removed.

**APPENDIX**

**THEOREM 1.** Given a system $S = \langle E, R \rangle$, a system $m_i = \langle E_{m_i}, R_{m_i} \rangle$ and a system $m_j = \langle E_{m_j}, R_{m_j} \rangle$. $S = m_i \cup m_j$ and $E_{m_i} \cap E_{m_j} = \emptyset \Rightarrow$ Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ and Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$.

**PROOF.**

$S = m_i \cup m_j \Rightarrow m_i \subseteq S$ and $m_j \subseteq S$ and $R_{m_i} \cup R_{m_j} = R$

$(\text{since} R_{m_i} \subseteq E_{m_i} \times E_{m_i} \text{and} R_{m_j} \subseteq E_{m_j} \times E_{m_j})$

$R_{m_i} \cap R_{m_j} = \emptyset$

implies that $R$ is the union of two disjoint sets: $R_{m_i}$ and $R_{m_j}$.

Since $(\text{for} m = m_i, m_j m = \langle E_{m_i}, R_{m_i} \rangle \subseteq S:\) Input$(R(m)) \cap R_{m} = \emptyset$ and Output$(R(m)) \cap R_{m} = \emptyset$, it holds that Input$(R(m))$, Output$(R(m))$, and Input$(R(m))$ are empty.

Hence, Input$(R(m)) \cap$ Output$(R(m)) = \emptyset$ and Input$(R(m)) \cap$ Output$(R(m)) = \emptyset$.

**THEOREM 2.** Given a system $S = \langle E, R \rangle$, a system $m_i = \langle E_{m_i}, R_{m_i} \rangle$ and a system $m_j = \langle E_{m_j}, R_{m_j} \rangle$ such that $m_i$ and $m_j$ are modules of $S$ and $E = E_{m_i} \cup E_{m_j}$:

$S = m_i \cup m_j \Rightarrow$ Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ and Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$.

**PROOF.**

1) $R_{m_i} \subseteq R$ and $R_{m_j} \subseteq R \Rightarrow R_{m_i} \cup R_{m_j} \subseteq R$

2) $\forall m \in \langle E_{m_i}, R_{m_i} \rangle \subseteq S:\$ Input$(R(m)) \subseteq R$ and Output$(R(m)) \subseteq R$

3) If Input$(R(m)) \cap$ Output$(R(m)) \neq \emptyset$ then $\exists <e_1, e_2> \in R: e_1 \in E_{m_i}$ and $e_2 \in E_{m_j}$ and $<e_1, e_2> \notin R_{m_i}$ and $<e_1, e_2> \notin R_{m_j}$

4) If Input$(R(m)) \cap$ Output$(R(m)) \neq \emptyset$ then $\exists <e_1, e_2> \in R: e_1 \in E_{m_i}$ and $e_2 \in E_{m_j}$ and $<e_1, e_2> \notin R_{m_i}$ and $<e_1, e_2> \notin R_{m_j}$

Hence, because of (1) to (4):

If Input$(R(m)) \cap$ Output$(R(m)) \neq \emptyset$ or Input$(R(m)) \cap$ Output$(R(m)) \neq \emptyset$, then $R_{m_i} \cup R_{m_j} \subseteq R$. However, if Input$(R(m)) \cap$ Output$(R(m)) = \emptyset$ and Input$(R(m)) \cap$ Output$(R(m)) = \emptyset$, then $R = R_{m_i} \cup R_{m_j}$.

Since $E_{m_i} \cup E_{m_j} = E$, Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ and Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ implies that $S = m_i \cup m_j$.

**THEOREM 3.** Given a system $S = \langle E, R \rangle$, a system $m_i = \langle E_{m_i}, R_{m_i} \rangle$ and a system $m_j = \langle E_{m_j}, R_{m_j} \rangle$ such that $m_i$ and $m_j$ are modules of $S$ and $E = E_{m_i} \cup E_{m_j}$:

$S = m_i \cup m_j$.

**PROOF.**

1) $\forall <e_1, e_2> \in R$ such that $e_1 \in E_{m_i}$ and $e_2 \in E - E_{m_i}$:

2) Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ or $e_1 \in E_{m_i}$ and $e_2 \in E_{m_j}$

Input$(R(m_i)) \cap$ Output$(R(m_j)) = \emptyset$ or $e_1 \in E_{m_i}$ and $e_2 \in E_{m_j}$

3) $\forall <e_1, e_2> \in R$ such that $e_1 \in E_{m_i}$ and $e_2 \in E_{m_j}$ and $(<e_1, e_2> \notin R_{m_i} \cup R_{m_j})$ or $(<e_1, e_2> \notin R_{m_i} \cup R_{m_j})$ or $(<e_1, e_2> \notin R_{m_i} \cup R_{m_j})$

4) $\forall <e_1, e_2> \in R: <e_1, e_2> \in R_{m_i} \cup R_{m_j}$ or $<e_1, e_2> \in R_{m_i} \cup R_{m_j}$ or $<e_1, e_2> \in R_{m_i} \cup R_{m_j}$

Given (1) to (4) it holds that:

$m_i \cup m_j = E_{m_i} \cup E_{m_j} R_{m_i} \cup R_{m_j} \Rightarrow$ Input$(R(m)) \cap$ Output$(R(m))$

Output$(R(m)) \cup$ Input$(R(m)) \cap$ Output$(R(m))\rangle = \langle E, R \rangle$

$\Rightarrow S$. 

**ACKNOWLEDGMENTS**

The authors thank Stuart H. Zweben, associate editor for TSE, and the anonymous referees for their valuable comments and corrections.

**REFERENCES**


