On Concept Algebra:
A Denotational Mathematical Structure for Knowledge and Software Modeling

Yingxu Wang, University of Calgary, Canada

ABSTRACT

Concepts are the most fundamental unit of cognition that carries certain meanings in expression, thinking, reasoning, and system modeling. In denotational mathematics, a concept is formally modeled as an abstract and dynamic mathematical structure that encapsulates attributes, objects, and relations. The most important property of an abstract concept is its adaptive capability to autonomously interrelate itself to other concepts. This article presents a formal theory for abstract concepts and knowledge manipulation known as "concept algebra." The mathematical models of concepts and knowledge are developed based on the object-attribute-relation (OAR) theory. The formal methodology for manipulating knowledge as a concept network is described. Case studies demonstrate that concept algebra provides a generic and formal knowledge manipulation means, which is capable to deal with complex knowledge and software structures as well as their algebraic operations.

Keywords: *case studies; cognitive informatics; concept; concept algebra; denotational mathematics; knowledge; knowledge network; learning; mathematics; natural intelligence; the OAR model*

INTRODUCTION

In cognitive informatics, logic, linguistics, psychology, software engineering, and knowledge engineering, concepts are identified as the basic unit of both knowledge and reasoning (Anderson, 1983; Colins & Loftus, 1975; Ganter & Wille, 1999; Hampton, 1997; Hurley, 1997; Matlin, 1998; Murphy, 1993; Wang, 2006a, 2006b, 2006c, 2007a, 2007c; Wang & Wang, 2006; Wilson & Keil, 1999). The rigorous modeling and formal treatment of concepts are at the center of theories for knowledge presentation and manipulation (Smith & Medin, 1981; Wille, 1982; Murphy, 1993; Codin, Missaoui, & Alaoui, 1995; Wilson & Keil, 1999; Yao, 2004; Chen & Yao, 2005). A *concept* in linguistics is a noun or noun-phrase that serves as the subject of a *to-be* statement (Hurley, 1997; Wang, 2002a, 2006a, 2006c, 2007d). Concepts in cognitive informatics (Wang, 2002a, 2006c, 2007b, 2007e) are an abstract structure that carries certain meaning in almost all cognitive processes such as thinking, learning, and reasoning.

Definition 1. *A concept is a cognitive unit to identify and/or model a real-world concrete entity and a perceived-world abstract subject.*

Based on concepts and their relations, meanings of real-world concrete entities may be represented and semantics of abstract subjects may be embodied. Concepts can be classified into two categories, known as the *concrete* and *abstract* concepts. The former are proper concepts that identify and model real-world entities such as the sun, a pen, and a computer. The latter are virtual concepts that identify and model abstract subjects, which cannot be directly mapped to a real-world entity, such as the mind, a set, and an idea. The abstract concepts may be further classified into *collective* concepts, such as collective nouns and complex concepts, or *attributive* concepts such as qualitative and quantitative adjectives. The concrete concepts are used to embody meanings of subjects in reasoning while the abstract concepts are used as intermediate representatives or modifiers in reasoning.

A concept can be identified by its intension and extension (Hurley, 1997; Smith & Medin, 1981; Wang, 2006c; Wille, 1982; Yao, 2004).

Definition 2. *The intension of a concept is the attributes or properties that a concept connotes.*

Definition 3. *The extension of a concept is the members or instances that the concept denotes.*

For example, the intension of the concept *pen* connotes the attributes of being a writing tool, with a nib, and with ink. The extension of the pen denotes all kinds of pens that share the common attributes as specified in the intension of the concept, such as a ballpoint pen, a fountain pen, and a quill pen.

In computing, a concept is an identifier or a name of a class. The intension of the class is a set of operational attributes of the class. The extension of the class is all its instantiations or objects and derived classes. Concept algebra provides a rigorous mathematical model and a formal semantics for object-oriented class modeling and analyses. The formal modeling of computational classes as a dynamic concept with predesigned behaviors may be referred to "system algebra" (Wang, 2006b, 2007d, 2008b, 2008d).

This article presents a formal treatment of abstract concepts and an entire set of algebraic operations on them. The mathematical model of concepts is established first. Then, the abstract mathematical structure, *concept algebra*, is developed for knowledge representation and manipulation. Based on concept algebra, a knowledge system is formally modeled as a concept network, where the methodology for knowledge manipulating is presented. Case studies demonstrate that concept algebra provides a denotational mathematical means for manipulating complicated abstract and concrete knowledge structures as well as their algebraic operations.

THE MATHEMATICAL MODEL OF ABSTRACT CONCEPTS

This section describes the formal treatment of abstract concepts and a new mathematical structure known as concept algebra in cognitive informatics and knowledge engineering. Before an abstract concept is defined, the semantic environment or context (Chen & Yao, 2005; Ganter & Wille, 1999; Hampton, 1997; Hurley, 1997; Medin & Shoben, 1988) in a given language, is introduced.

Definition 4. *Let $\mathcal{O}$ denote a finite or infinite nonempty set of objects, and $\mathcal{A}$ be a finite or infinite nonempty set of attributes, then a semantic environment or context Θ is denoted as a triple, i.e.:*

$$\begin{aligned}\Theta &\triangleq (\mathcal{O}, \mathcal{A}, \mathcal{R}) \\ &= \mathcal{R}: \mathcal{O} \to \mathcal{O} \,|\, \mathcal{O} \to \mathcal{A} \,|\, \mathcal{A} \to \mathcal{O} \,|\, \mathcal{A} \to \mathcal{A}\end{aligned} \quad (1)$$

where $\mathcal{R}$ is a set of relations between $\mathcal{O}$ and $\mathcal{A}$, and | demotes alternative relations.

According to the Object-Attribute-Relation (OAR) model (Wang, 2007c, 2007d; Wang & Wang, 2006), the three essences in Θ can be defined as follows.

Definition 5. *An object o is an instantiation of a concrete entity and/or an abstract concept.*

In a narrow sense, an object is the identifier of a given instantiation of a concept.

Definition 6. *An attribute a is a subconcept that is used to characterize the properties of a given concept by more specific or precise concepts in the abstract hierarchy.*

In a narrow sense, an attribute is the identifier of a subconcept of the given concept.

Definition *7. A relation r is an association between any pair of object-object, object-attribute, attribute-object, and/or attribute-attribute.*

On the basis of OAR and Θ, an abstract concept is a composition of the above three elements as given below.

Definition 8. *An abstract concept c on Θ is a 5-tuple, i.e.:*

$$c \triangleq (O, A, R^c, R^i, R^o) \qquad (2)$$

where

- O is a nonempty set of objects of the concept, $O = \{o_1, o_2, ..., o_m\} \subseteq \text{Þ}\mathcal{O}$, where $\text{Þ}\mathcal{O}$ denotes a power set of $\mathcal{O}$.
- A is a nonempty set of attributes, $A = \{a_1, a_2, ..., a_n\} \subseteq \text{Þ}\mathcal{A}$.
- $R^c = O \times A$ is a set of internal relations.
- $R^i \subseteq C' \times C$ is a set of input relations, where C′ is a set of external concepts.
- $R^o \subseteq C \times C'$ is a set of output relations.

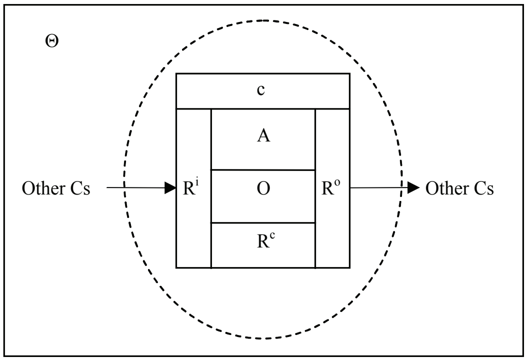
The structure of the concept model $c = (O, A, R^c, R^i, R^o)$ can be illustrated in Figure 1, where $c, A, O,$ and R, $R = \{R^c, R^i, R^o\}$, denote the identifier of the concept, its attributes, objects, and internal/external relations, respectively.

It is interesting to compare the formal model of abstract concepts as given in Definition 8 with the notion of the concept lattice proposed by Wille (1982). Wille defined a formal concept as a pair of sets of objects and attributes, i.e.:

$$c \triangleq (O, A),\ O \subseteq \mathcal{O} \wedge A \subseteq \mathcal{A} \qquad (3)$$

It is obvious that the abstract concept extends Wille's concept model from a pair to a triple, where the set of relations is explicitly and formally modeled in three categories known as the internal, input, and output relations. The I/O relations enable a conventional static concept to be dynamically associated to other concepts in order to represent and manipulate complicated concept operations and compositions in a concept network and knowledge hierarchy.

Figure 1. The structured model of an abstract concept



Theorem 1. *The dynamic and adaptive property of concepts states that an abstract concept is a dynamic mathematical structure that possesses the adaptive capability to interrelate itself to other concepts via* R^i *and* R^o.

Based on Definition 8, an object derived from a concept and the intension/extension of a concept can be formally defined as follows.

Definition 9. *An object of a concept o is a derived instantiation of the concept that implements an end product of the concept,* $o \subset O$, *i.e.:*

$$\forall c(O, A, R^c, R^i, R^o), o = c.o_i, o_i \subset O, R^o{}_o \equiv \varnothing \Rightarrow o(A_o, R^c{}_o, R^i{}_o | A_o \supseteq A, R^c{}_o = o \times A_o, R^i{}_o = \{(c, o)\}) \quad (4)$$

Equation 4 indicates that an object is a tailored end-product of a concrete concept where there is no any output-oriented relation to another concept.

Definition 10. *The intension of a concept c = (O, A,* R^c, R^i, R^o*), c*, is represented by its set of attributes A, i.e.:*

$$c^*(O, A, R^c, R^i, R^o) \triangleq A = \bigcap_{i=1}^{\#O} A_{o_j} \subseteq Þ\mathcal{A} \quad (5)$$

where $Þ\mathcal{A}$ denotes a power set of $\mathcal{A}$, and # is the cardinal operator that counts the number of elements in a given set.

Definition 10 indicates that the *narrow* sense or the exact semantics of a concept is determined by the set of common attributes shared by all of its objects. In contrary, the *broad* sense or the rough semantics of a concept is referred to the set of all attributes identified by any of its objects as defined below.

Definition 11. *The complete set of attributes of a concept c = (O, A,* R^c, R^i, R^o*), or the instant attributes denoted by all objects of c, is a closure of all objects' intensions, A*, i.e.:*

$$A^* \triangleq \bigcup_{j=1}^{\#O} A_{o_j} \quad (6)$$

Definitions 10 and 11 specify that (a) The intension of a concept is a finite set of objectively identifiable attributes at a given level of abstraction, and (b) the intension of a concept is dynamic. When more objects for the same concept are denoted, the domain of the intension is usually shrinking in order to accommodate the new objects in the same structure of the concept.

Conventionally, the *domain* of a concept's intension is used to be perceived subjectively in literature (Hurley, 1997; Matlin, 1998). In this approach, it is deemed that a concept connotes the attributes, which occur in the minds of people who use that concept, or where something must have in order to be denoted by the concept. Both the above informal perceptions are not objectively operational in defining a complete and unambiguity domain of intensions. To solve this fundamental problem, Definition 10 provides a unique and objective determination of any given concept.

Definition 12. *The extension of a concept c = (O, A,* R^c, R^i, R^o*),* c^+*, is represented by its set of objects O, i.e.:*

$$c^+(O, A, R^c, R^i, R^o) \triangleq O = \{o_1, o_2, ..., o_m\} \subseteq Þ\mathcal{O} \quad (7)$$

A formal and objective definition of the domain of intension is provided below.

Definition 13. *The domain of a concept c = (O, A,* R^c, R^i, R^o*) is a set of attributes with a narrow sense* D_{min} *referring to its intension and a broad sense* D_{max} *referring to its closure, i.e.:*

$$D(c) \triangleq \begin{cases} D_{\min}(c) = A = \bigcap_{j=1}^{\#O} A_{o_j} \\ D_{\max}(c) = A^* = \bigcup_{j=1}^{\#O} A_{o_j} \end{cases} \quad (8)$$

It is noteworthy that in conventional literature, it is only believed that the intension of a concept determines its extension (Hurley, 1997; Matlin, 1998). However, Definition 13 reveals that the extension of a concept, particularly the common attributes elicited from the extension, determines its intension as well.

Theorem 2. *The nature of concept hierarchy states that in an abstraction hierarchy, the higher the level of a concept in abstraction, the smaller the intension of the concept; and vice versa.*

Relationships between concepts in a concept hierarchy can be illustrated in Figure 2 at three levels known as the knowledge, object, and attribute levels. The internal relations of concepts, $R^c = O \times A$ can be formally represented by concept matrixes.

Example 1. *The concept matrix of concept c_1 is given in Table 1. According to Definition 13, the intension and extension of concepts c_1 as specified in Table 1 can be objectively and uniquely determined as: $A_{c1} = \{a_2, a_4\}$ and $O_{c1} = \{o_{11}, o_{12}, o_{13}\}$.*

Definition 14. *The identification of a new concept $c(O, A, R^c, R^i, R^o)$ is the elicitation of its objects O, attributes A, and internal relations R^c, from the semantic environment $\Theta = (\mathcal{O}, \mathcal{A}, \mathcal{R})$, i.e.:*

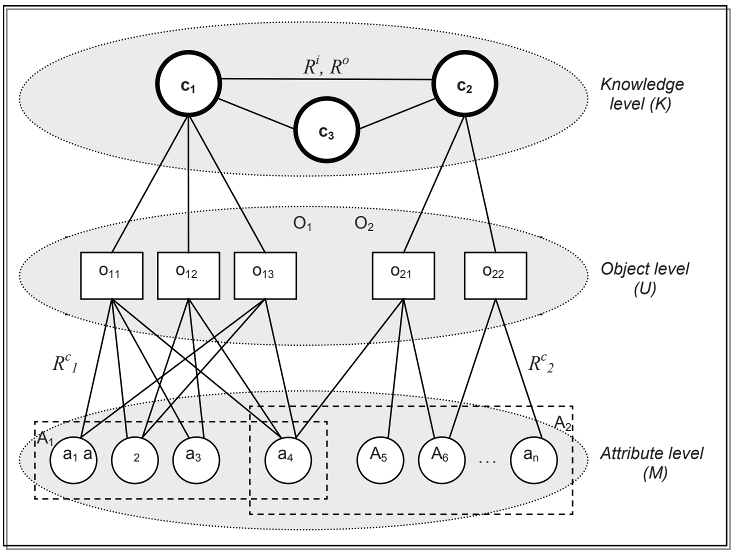
$$c \triangleq (O, A, R^c, R^i, R^o \mid O \subset U \wedge A \subset M, R^c = O \times A, R^i = \varnothing, R^c = \varnothing) \quad (9)$$

In Definition 14, $R^i = R^o = \varnothing$ denotes that the identification operation is an initialization

Table 1. The concept matrix of c_1

c_1	a_1	a_2	a_3	a_4
o_{11}	1	1	1	1
o_{12}		1	1	1
o_{13}	1	1		1

Figure 2. The hierarchical relations of concepts and their internal structures



of a newly created concept where the input and output relations may be established later.

Definition 15. *A qualification of a concept c(O, A, R^c, R^i, R^o), denoted by *c, is the identification of its domain, i.e.:*

$$*c \triangleq D(c) = \begin{cases} D_{\min}(c) = A = \bigcap_{j=1}^{\#O} A_{o_j} \\ D_{\max}(c) = A^* = \bigcup_{j=1}^{\#O} A_{o_j} \end{cases} \quad (10)$$

Definition 16. *A quantification of a concept c(O, A, R^c, R^i, R^o), denoted by #c, is the cardinal evaluation of its domain in term of the number of attributes included in it, i.e.:*

$$\#c \triangleq \#D(c) = \begin{cases} \#D_{\min}(c) = \#A = \#(\bigcap_{j=1}^{\#O} A_{o_j}) \\ \#D_{\max}(c) = \#A^* = \#(\bigcup_{j=1}^{\#O} A_{o_j}) \end{cases} \quad (11)$$

Example 2. *According to Definitions 15 and 16, the qualification and quantification of concept c_1 as given in Figure 2 are as follows, respectively:*

$$*c_1 = D(c_1) = \begin{cases} D_{\min}(c_1) = A_{c_1} = \{a_2, a_4\} \\ D_{\max}(c_1) = A_{c_1}^* = \{a_1, a_2, a_3, a_4\} \end{cases}$$

$$\#c_1 = \#D(c_1) = \begin{cases} \#D_{\min}(c_1) = \#A_{c_1} = 2 \\ \#D_{\max}(c_1) = \#A_{c_1}^* = 4 \end{cases}$$

Concept algebra is an abstract mathematical structure for the formal treatment of concepts and their algebraic relations, operations, and associative rules for composing complex concepts.

Definition 17. *A concept algebra CA on a given semantic environment Θ is a triple, i.e.:*

$$CA \triangleq (C, OP, \Theta) = (\{O, A, R^c, R^i, R^o\}, \{\bullet_r, \bullet_c\}, \Theta) \quad (12)$$

where OP = {$\bullet_r$, $\bullet_c$}are the sets of relational and compositional operations on abstract concepts.

Concept algebra provides a denotational mathematical means for algebraic manipulations of abstract concepts. Concept algebra can be used to model, specify, and manipulate generic "to be" type problems, particularly system architectures, knowledge bases, and detail-level system designs in computing, software engineering, system engineering, and cognitive informatics. The relational and compositional operations on concepts will be formally described in the following sections.

RELATIONAL OPERATIONS OF CONCEPTS

The relational operations of abstract concepts are static and comparative operations that do not change the concepts involved. It is recognized that *relationships* between concepts are solely determined by the relations of both their intensions *A* and extensions *O*. The relational operations on abstract concepts in concept algebra are described below.

Lemma 1. *The relational operations $\bullet_r$ in concept algebra encompasses 8 comparative operators for manipulating the algebraic relations between concepts, i.e.:*

$$\bullet_r \triangleq \{\leftrightarrow, \nleftrightarrow, \prec, \succ, =, \cong, \sim, \triangleq\} \quad (13)$$

where the relational operators stand for *related, independent, subconcept, superconcept, equivalent, consistent, comparison,* and *definition*, respectively.

Definition 18. *The related concepts c_1 and c_2 on Θ, denoted by ↔, are a pair of concepts that share some common attributes in their intensions A_1 and A_2, i.e.:*

$$c_1 \leftrightarrow c_2 \triangleq A_1 \cap A_2 \neq \varnothing \quad (14)$$

Definition 19. *The independent concepts c_1 and c_2 on Θ, denoted by $\not\leftrightarrow$, are two concepts that their intensions A_1 and A_2 are disjoint, i.e.:*

$$c_1 \not\leftrightarrow c_2 \triangleq A_1 \cap A_2 = \varnothing \quad (15)$$

It is obvious that related and independent concepts are mutually exclusive. That is, if $c_1 \leftrightarrow c_2$, then $\neg (c_1 \not\leftrightarrow c_2)$; and vice versa.

Definition 20. *A subconcept c_1 of concept c_2 on Θ, denoted by $\prec$, is a concept that its intension A_1 is a superset of A_2, i.e.:*

$$c_1 \prec c_2 \triangleq A_1 \supset A_2 \quad (16)$$

Definition 21. *A superconcept c_2 over concept c_1 on Θ, denoted by $\succ$, is a concept that its intension A_2 is a subset of A_1, i.e.:*

$$c_2 \succ c_1 \triangleq A_2 \subset A_1 \quad (17)$$

According to Definitions 20 and 21, a subconcept and a superconcept are reflective. That is, *if* $c_1 \prec c_2$, then $c_2 \succ c_1$.

Definition 22. *The equivalent concepts c_1 and c_2 on Θ, denoted by =, are two concepts that their intensions (A_1, A_2), and extensions (O_1, O_2) are identical, i.e.:*

$$c_1 = c_2 \triangleq (A_1 = A_2) \wedge (O_1 = O_2) \quad (18)$$

Definition 23. *The consistent concepts c_1 and c_2 on Θ, denoted by $\cong$, are two concepts with a relation of being either a sub- or superconcept, i.e.:*

$$c_1 \cong c_2 \triangleq (c_1 \succ c_1) \vee (c_1 \prec c_2) = (A_1 \subset A_1) \vee (A_1 \supset A_2) \quad (19)$$

Definition 24. *A comparison between two concepts c_1 and c_2 on Θ, denoted by ~, is an operation that determines the equivalency or similarity level of their intensions, i.e.:*

$$c_1 \sim c_2 \triangleq \frac{\#(A_1 \cap A_2)}{\#(A_1 \cup A_2)} * 100\% \quad (20)$$

The range of equivalency between two concepts is among 0 to 100 %, where 0% means no similarity and 100% means a full similarity. According to Definition 24, It is obvious that:

$$c_1 \sim c_2 = \begin{cases} 100\%, & c_1 = c_2 \\ \frac{\#A_2}{\#A_1} \bullet 100\%, & c_1 \prec c_2 \\ \frac{\#A_1}{\#A_2} \bullet 100\%, & c_1 \succ c_2 \end{cases} \quad (21)$$

Definition 25. *A definition of a concept c_1 by c_2 on Θ, denoted by $\triangleq$, is an association between two concepts where they are equivalent, i.e.:*

$$c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2)$$
$$\triangleq c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1 \mid O_1 = O_2, A_1 = A_2, R^c{}_1 = O_1 \times A_1, R^i{}_1 = R^i{}_2, R^o{}_1 = R^o{}_2) \quad (22)$$

COMPOSITIONAL OPERATIONS OF CONCEPTS

The compositional operations of concept algebra are dynamic and integrative operations that always change all concepts involved in parallel. Compositional operations on concepts provide a set of fundamental mathematical means to construct complex concepts on the basis of simple ones or to derive new concepts on the basis of exiting ones.

Lemma 2. *The compositional operations $\bullet_c$ in concept algebra encompasses nine associative operators for manipulating the algebraic compositions among concepts, i.e.:*

$$\bullet_c \triangleq \{\Rightarrow, \overset{-}{\Rightarrow}, \overset{+}{\Rightarrow}, \overset{\sim}{\Rightarrow}, \uplus, \pitchfork, \Lleftarrow, \vdash, \mapsto\} \quad (23)$$

where the compositional operators stand for *inheritance, tailoring, extension, substitute, composition, decomposition, aggregation, specification,* and *instantiation*, respectively.

The compositional operations of concept algebra can be illustrated in Figure. 3. In Figure 3, $R = \{R^c, R^i, R^o\}$, and all nine compositional operations define composing rules among concepts, except instantiation that is an operation between a concept and a specific object.

Definition 26. *An inheritance of concept c_2 from concept c_1, denoted by $\Rightarrow$, is the creation of the new concept c_2 by reproducing c_1, and the establishment of new associations between them in parallel, see Box 1, where c_1 is called the parent concept, c_2 is the child concept, and || denotes that an inheritance creates new associations between c_1 and c_2 in parallel via ($R^o{}_1$, $R^i{}_2$) and ($R^o{}_2$, $R^i{}_1$).*

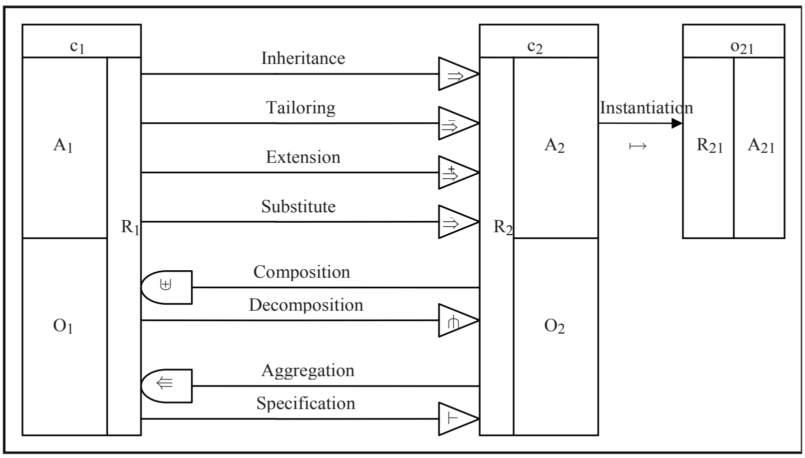
Definition 27. *The multiple inheritance of concept c from n parent concepts c_1, c_2, ..., c_n, denoted by $\Rightarrow$, is an inheritance that creates the new concept c via a set of n conjoint concepts and establishes new associations among them, see Box 2, where $\overset{n}{\underset{i=1}{R}}(x_i)$ is known as the big-R notation (Wang, 2002b, 2008c) that denotes a repetitive behavior or recurrent structure.*

Definition 28. *The tailoring of concept c_2 from the parent concept c_1, denoted by $\overset{-}{\Rightarrow}$, is a special inheritance that creates the new concept c_2 based on c_1 with the removal of the subsets of objects O' and attributes A'; at the same time, it establishes new associations between the two concepts, see Box 3.*

Definition 29. *The extension of concept c_2 from the parent concept c_1, denoted by $\overset{+}{\Rightarrow}$, is a special inheritance that creates the new concept c_2 based c_1 with additional objects O' and/or attributes A', and establishes new associations between the two concepts, see Box 4.*

Definition 30. *The substitute of concept c_2 from the parent concept c_1, denoted by $\overset{\sim}{\Rightarrow}$, is a special inheritance that creates the new concept c_2 based on c_1 by replacing the inherited subsets of objects O'_{c1} and attributes A'_{c1} with corresponding ones O'_{c2} and A'_{c2} that share the*

Figure 3. The compositional operations of concept algebra as concept manipulation rules



Box 1.

$$
\begin{aligned}
& c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \Rightarrow c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2) \\
& \triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2 \mid O_2 = O_1, A_2 = A_1, R^c{}_2 = O_2 \times A_2, \\
& \quad \{R^i{}_2 = R^i{}_1 \cup \ (c_1, c_2)\}, R^o{}_2 = R^o{}_1 \cup \{(c_2, c_1)\}\}) \\
& \mid\mid c_1(O_1, A_1, R^c{}_1, R^{i'}{}_1, R^{o'}{}_1 \mid R^{i'}{}_1 = R^i{}_1 \cup \{(c_2, c_1)\}, R^{o'}{}_1 = R^o{}_1 \cup \{(c_1, c_2)\}\})
\end{aligned}
\tag{24}
$$

Box 2.

$$
\begin{aligned}
& \mathop{R}_{i=1}^{n} (c_i \Rightarrow c\ O, A, R^c, R^i, R^o) \\
& \triangleq c(O, A, R^c, R^i, R^o \mid O = \bigcup_{i=1}^{n} O_{c_i}, A = \bigcup_{i=1}^{n} A_{c_i}, R^c = O \times A, \\
& \quad R^i = \bigcup_{i=1}^{n} R^i{}_{c_i} \cup \{\mathop{R}_{i=1}^{n} (c_i, c)\}, R^o = \bigcup_{i=1}^{n} R^o{}_{c_i} \cup \{\mathop{R}_{i=1}^{n} (c, c_i)\}\}) \\
& \mid\mid \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}\})
\end{aligned}
\tag{25}
$$

Box 3.

$$
\begin{aligned}
& c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \overset{-}{\Rightarrow} c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2) \\
& \triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2 \mid O_2 = O_1 \setminus O', A_2 = A_1 \setminus A', R^c{}_2 = O_2 \times A_2 \subset R^c{}_1, \\
& \quad R^i{}_2 = R^i{}_1 \cup \{(c_1, c_2)\}, R^o{}_2 = R^o{}_1 \cup \{(c_2, c_1)\}\}) \\
& \mid\mid c_1(O_1, A_1, R^c{}_1, R^{i'}{}_1, R^{o'}{}_1 \mid R^{i'}{}_1 = R^i{}_1 \cup \{(c_2, c_1)\}, R^{o'}{}_1 = R^o{}_1 \cup \{(c_1, c_2)\}\})
\end{aligned}
\tag{26}
$$

Box 4.

$$
\begin{aligned}
& c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \overset{+}{\Longrightarrow} c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2) \\
& \triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2 \mid O_2 = O_1 \cup O', A_2 = A_1 \cup A', R^c{}_2 = O_2 \times A_2 \supset R^c{}_1, \\
& \quad R^i{}_2 = R^i{}_1 \cup \{(c_1, c_2)\}, R^o{}_2 = R^o{}_1 \cup \{(c_2, c_1)\}\}) \\
& \mid\mid c_1(O_1, A_1, R^c{}_1, R^{i'}{}_1, R^{o'}{}_1 \mid R^{i'}{}_1 = R^i{}_1 \cup \{(c_2, c_1)\}, R^{o'}{}_1 = R^o{}_1 \cup \{(c_1, c_2)\}\})
\end{aligned}
\tag{27}
$$

Box 5.

$$c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \ \tilde{\Rightarrow} \ c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2)$$
$$\triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2 \mid O_2 = (O_1 \setminus O'_{c_1}) \cup O'_{c_2}, A_2 = (A_1 \setminus A'_{c_1}) \cup A'_{c_2},$$
$$R^c{}_2 = O_2 \times A_2, R^i{}_2 = R^i{}_1 \cup \{(c_1, c_2)\}, R^o{}_2 = R^o{}_1 \cup \{(c_2, c_1)\})$$
$$|| \, c_1(O_1, A_1, R^c{}_1, R^{i'}{}_1, R^{o'}{}_1 \mid R^{i'}{}_1 = R^i{}_1 \cup \{(c_2, c_1)\}, R^{o'}{}_1 = R^o{}_1 \cup \{(c_1, c_2)\}) \qquad (28)$$

same identifiers but possess different objects or attributes; at the same time, it establishes new associations between the two concepts, see Box 5, where $O'_{c1} \subset O_1 \wedge O'_{c2} \subset O_2 \wedge \#O'_{c1} = \#O'_{c2}$ *and* $A'_{c1} \subset A_1 \wedge A'_{c2} \subset A_2 \wedge \#A'_{c1} = \#A'_{c2}$.

Binary concept tailoring, extension, and substitution can be extended to corresponding *n*-nary operations, similar to that of inheritance as given in Definition 27.

Definition 31. *The composition of concept c from n subconcepts* $c_1, c_2, \ldots, c_n$*, denoted by* $\uplus$*, is an integration of them that creates the new super concept c via concept conjunction; at the same time, it establishes new associations between them, see Box 6.*

It is noteworthy that, according to the calculus of incremental union $\boxplus$ (Wang, 2006b, 2008b), the composition operation as given in Definition 31 results in the generation of new internal relations, which do not belong to any of its subconcepts. This is the most important property of concept composition.

Corollary 1. *The composition of multiple concepts is an incremental union operation, where the newly generated internal relations* ΔR^c *can be determined as:*

$$\Delta R^c = \bigcup_{i=1}^{n} \{(c, c_i), (c_i, c)\} \qquad (30)$$

A concept decomposition is an inverse operation of concept compositions.

Definition 32. *The decomposition of concept c into n subconcepts* $c_1, c_2, \ldots, c_n$*, denoted by* $\pitchfork$*, is a partition of the superconcept into multiple*

Box 6.

$$c(O, A, R^c, R^i, R^o) \triangleq \biguplus_{i=1}^{n} c_i(O_i, A_i, R_i^c, R_i^i, R_i^o)$$
$$= c(O, A, R^c, R^i, R^o \mid O = \bigcup_{i=1}^{n} O_{c_i}, A = \bigcup_{i=1}^{n} A_{c_i}, R^c = \bigcup_{i=1}^{n} R^c{}_{c_i} \cup \{(c, c_i), (c_i, c)\},$$
$$R^i = \bigcup_{i=1}^{n} R^i{}_{c_i} \ \ R^o = \bigcup_{i=1}^{n} R^o{}_{c_i})$$
$$|| \, \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \qquad (29)$$

Box 7.

$$
\begin{aligned}
& c(O, A, R^c, R^i, R^o) \pitchfork \overset{n}{\underset{i=1}{R}} c_i \\
& \triangleq \overset{n}{\underset{i=1}{R}} \{ c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \\
& \qquad || \, c(O, A, R^{c'}, R^{i'}, R^{o'} \mid R^{c'} = \bigcup_{i=1}^{n} (R^c{}_{c_i} \setminus \{(c, c_i)(c_i, c)\}), \\
& \qquad\quad \{R^{i'} = R^i \cup \overset{n}{\underset{i=1}{R}} (c_i, c)\}, R^{o'} = R^o \cup \{\overset{n}{\underset{i=1}{R}} (c, c_i)\}) \\
& \qquad\quad \setminus \; c_i(O_i, A_i, R^c{}_i, R^i{}_i, R^o{}_i) \\
& \qquad \}
\end{aligned}
\tag{31}
$$

subconcepts; at the same time, it establishes new associations between them, see Box 7.

As specified in Definition 32, the decomposition operation results in the removal of all internal relations $\Delta R^c = \bigcup_{i=1}^{n} \{(c, c_i), (c_i, c)\}$ that are no longer belong to any of its subconcepts.

Definition 33. *The aggregation of concept c_1 from concept c_2, denoted by $\Lleftarrow$, is a creation of c_1 via abstraction of c_2 with a reduced intension of more generic attributes; at the same time, it establishes new associations between them, see Box 8.*

Concept aggregation is also known as concept *generalization, abstraction,* or *elicitation*. Binary aggregations can be extended to *n*-nary parallel or serial aggregations.

Definition 34. *The parallel aggregation of a concept c from a set of n concepts c_1, c_2, ..., c_n, denoted by $\Lleftarrow$, is an aggregation of c with the elicitation of all concepts in the set, see Box 9.*

Based on Definition 34, a concept may be inductively generalized by a series of aggregations with a smaller set of more abstract (super) attributes.

Definition 35. *The serial aggregation of a concept c from a set of n concepts c_1, c_2, ..., c_n, denoted by $\Lleftarrow$, is an aggregation with a total order of a series of decreasing intensions of the concepts by more abstract and generic attributes, see Box 10.*

Box 8.

$$
\begin{aligned}
& c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \;\Lleftarrow\; c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2) \\
& \triangleq c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1 \mid O_1 \supset O_2, A_1 \subset A_2, R^c{}_1 = (O_1 \times A_1) \cup \\
& \qquad \{(c_1, c_2), (c_2, c_1)\}, R^i{}_1 = R^i{}_2 \cup \{(c_2, c_1)\}, R^o{}_1 = R^o{}_2 \cup \{(c_1, c_2)\}) \\
& || \, c_2(O_2, A_2, R^c{}_2, R^{i'}{}_2, R^{o'}{}_2 \mid R^{i'}{}_2 = R^i{}_2 \cup \{(c_1, c_2)\}, R^{o'}{}_2 = R^o{}_2 \cup \{(c_2, c_1)\})
\end{aligned}
\tag{32}
$$

Box 9.

$$c(O, A, R^c, R^i, R^o) \Leftarrow \mathop{R}_{i=1}^{n} c_i$$
$$\triangleq c(O, A, R^c, R^i, R^o \mid O = \bigcup_{i=1}^{n} O_i, A = \bigcap_{i=1}^{n} A_i, R^c = \bigcup_{i=1}^{n} (R^c{}_{c_i} \cup \{(c, c_i), (c_i, c)\}),$$
$$(R^i = \bigcup_{i=1}^{n} R^i{}_{c_i} \cup \{(c_i, c)\}), R^o = \bigcup_{i=1}^{n} (R^o{}_{c_i} \cup \{(c, c_i)\}),$$
$$|| \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \quad (33)$$

Box 10.

$$c(O, A, R^c, R^i, R^o) \Leftarrow (c_1 \Leftarrow c_2 \Leftarrow ... \Leftarrow c_n)$$
$$\triangleq c(O, A, R^c, R^i, R^o \mid O \supset O_1 \supset O_2 \supset ... \supset O_n, A \subset A_1 \subset A_2 \subset ... \subset A_n,$$
$$R^c = (O \times A) \cup \{(c, c_1), (c_1, c)\}, R^i = R^i{}_1 \cup \{\mathop{R}_{i=1}^{n} (c_i, c)\},$$
$$R^o = R^o{}_1 \cup \{\mathop{R}_{i=1}^{n} (c, c_i)\})$$
$$|| \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \quad (34)$$

A concept specification is an inverse operation of concept aggregations.

Definition 36. *The specification of concept c_1 by concept c_2, denoted by $\vdash$, is a deductive refinement of c_1 by an increasing intension with more specific and precise attributes in c_2; at the same time, it establishes new associations between them, see Box 11.*

Binary specifications can be extended to *n*-nary parallel or serial aggregations.

Definition 37. *The parallel specification of concept c by a set of n concepts c_1, c_2, ..., c_n, denoted by $\vdash$, is a specification of c with the elicitation of all concepts in the set, see Box 12.*

Definition 38. *The serial specification of concept c by a set of concepts c_1, c_2, ..., c_n, denoted by $\vdash$, is a specification with a total order of a series of refinements by increasing intensions of the concepts with more specific and precise attributes, see Box 13.*

The *binary, parallel,* and *series* specifications and aggregations of concepts provide a generic means for forming a hierarchical structure of concepts in knowledge engineering.

Theorem 3. *A totally ordered series of decreasing intensions in a serial concept aggregation is reversely proportional to a totally ordered series of increasing extensions, i.e.:*

Box 11.

$$c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \vdash c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2)$$
$$\triangleq c_2(O_2, A_2, R^c{}_2, R^i{}_2, R^o{}_2 \mid O_2 \subset O_1, A_2 \supset A_1, R^c{}_2 = (O_2 \times A_2) \cup$$
$$\{(c_2, c_1), (c_1, c_2)\}, R^i{}_2 = R^i{}_1 \cup \{(c_1, c_2)\}, R^o{}_2 = R^o{}_1 \cup \{(c_2, c_1)\})$$
$$\| c_1(O_1, A_1, R^c{}_1, R^{i'}{}_1, R^{o'}{}_1 \mid R^{i'}{}_1 = R^i{}_1 \cup \{(c, c_1)\}, R^{o'}{}_1 = R^o{}_1 \cup \{(c_1, c)\}) \tag{35}$$

Box 12.

$$\mathop{R}_{i=1}^{n} c_i \vdash c(O, A, R^c, R^i, R^o)$$
$$\triangleq c(O, A, R^c, R^i, R^o \mid O \subset \bigcup_{i=1}^{n} O_i, A = \bigcap_{i=1}^{n} A_i, R^c = \bigcup_{i=1}^{n} (R^c{}_{c_i} \cup \{(c, c_i), (c_i, c)\}),$$
$$R^i = \bigcup_{i=1}^{n} (R^i{}_{c_i} \cup \{(c_i, c)\}), R^o = \bigcup_{i=1}^{n} (R^o{}_{c_i} \cup \{(c, c_i)\}))$$
$$\| \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \tag{36}$$

Box 13.

$$(c_n \vdash \ldots \vdash c_2 \vdash c_1) \vdash c(O, A, R^c, R^i, R^o)$$
$$\triangleq c(O, A, R^c, R^i, R^o \mid O_n \supset \ldots \supset O_2 \supset O_1 \supset O, A_n \subset \ldots \subset A_2 \subset A_1 \subset A,$$
$$R^c = (O \times A) \cup \mathop{R}_{i=1}^{n} \{(c, c_i), (c_i, c)\},$$
$$R^i = R^i{}_n \cup \mathop{R}_{i=1}^{n} \{(c_i, c)\}), R^o = R^o{}_n \cup \mathop{R}_{i=1}^{n} \{(c, c_i)\})$$
$$\| \mathop{R}_{i=1}^{n} c_i(O_i, A_i, R^c{}_i, R^{i'}{}_i, R^{o'}{}_i \mid R^{i'}{}_i = R^i{}_i \cup \{(c, c_i)\}, R^{o'}{}_i = R^o{}_i \cup \{(c_i, c)\}) \tag{37}$$

$$\forall c \Lleftarrow c_1 \Lleftarrow c_2 \Lleftarrow \ldots \Lleftarrow c_n,$$
$$A \subset A_1 \subset A_2 \subset \ldots \subset A_n \Rightarrow O \supset O_1 \supset O_2 \supset \ldots \supset O_n \tag{38}$$

Example 3. *The relationships between series of specifications and aggregations on the concept animal can be described as follows:*

$$(animal \vdash mammal \vdash feline \vdash tiger) \Rightarrow$$
$$(animal \Lleftarrow mammal \Lleftarrow feline \Lleftarrow tiger)$$

where, according Theorem 3, it can be obtained:

$A_{animal} \subset A_{mammal} \subset A_{feline} \subset A_{tiger}$, and
$O_{animal} \supset O_{mammal} \supset O_{feline} \supset O_{tiger}$

The compositional operations of concepts formally defined so far are those among abstract concepts. The remainder of this section describes a special compositional operation between a given concept and its objects.

Definition 39. *The instantiation of a concept c, denoted by ↦, is an embodiment of its generic semantics onto a specific case or implementation known as an object o, see Box 14.*

It is noteworthy that the output relation of an object is always empty (i.e., $R^o{}_o \equiv \varnothing$), which means that the object is an end product of a concept where there is no further deduction of meanings in the hierarchy of the inheritance chain.

Definition 40. *The multiple instantiation of a concept c onto n objects, denoted by ↦, is a compound parallel instantiation that creates a set of new objects $\{o_1, o_2, ..., o_n\}$ based on c, and establishes new associations between them, see Box 15.*

CONCEPT ALGEBRA FOR KNOWLEDGE MANIPULATION

This section describes applications of concept algebra in the manipulation of abstract models of knowledge and the methodology for knowledge representation and manipulation.

The Mathematical Model of Knowledge

In cognitive informatics (Wang, 2002a, 2006a, 2007e), particularly the OAR model (Wang, 2007c; Wang & Wang, 2006) on internal knowledge representation in the brain, human *knowledge* is modeled as a concept network, where concept algebra is applied as a set of rules for knowledge composition in order to construct complex and dynamic concept networks.

Definition 41. *A generic knowledge K is an n-nary relation $\Re$ among a set of n concepts and the entire set of concepts C, i.e.:*

$$K = \Re : (\mathop{\mathsf{X}}_{i=1}^{n} C_i) \to C \tag{41}$$

where $\biguplus_{i=1}^{n} C_i = C$*, and*

Box 14.

$$\begin{aligned} &c(O, A, R^c, R^i, R^o) \;\mapsto\; o(A_o, R^c{}_o, R^i{}_o) \\ &\triangleq o(A_o, R^c{}_o, R^{i'}{}_o \mid o \subset O, A_o = A, R^c{}_o = o \times A_o, R^{i'}{}_o = R^i{}_o \cup \{(c,o)\}) \\ &\;||\, c(O, A, R^c, R^{i'}, R^{o'} \mid R^{i'} = R^i \cup \{(o,c)\}, R^{o'} = R^o \cup \{(c,o)\}) \end{aligned} \tag{39}$$

Box 15.

$$\begin{aligned} &c(O, A, R^c, R^i, R^o) \;\mapsto\; \mathop{R}_{i=1}^{n} o_i(A_{o_i}, R^c{}_{o_i}, R^i{}_{o_i}) \\ &\triangleq \mathop{R}_{i=1}^{n} o_i(A_{o_i}, R^c{}_{o_i}, R^{i'}{}_{o_i} \mid o_i \subseteq O, A_{o_i} = A, R^c{}_{o_i} = o_i \times A_{o_i}, \\ &\qquad R^{i'}{}_{o_i} = R^i{}_{o_i} \cup \; (c, o_i)\}) \\ &\;||\, c(O, A, R^c, R^{i'}, R^{o'} \mid R^{i'} = R^i \cup \{\mathop{R}_{i=1}^{n}(o_i, c)\}, R^{o'} = R^o \cup \{\mathop{R}_{i=1}^{n}(c, o_i)\}) \end{aligned} \tag{40}$$

$$\Re = \bullet_c = \{\Rightarrow, \overset{-}{\Rightarrow}, \overset{+}{\Rightarrow}, \overset{\sim}{\Rightarrow}, \uplus, \pitchfork, \Lleftarrow, \vdash, \mapsto\}.$$

According to Definition 41, the most simple knowledge k is a binary relation $\Re$ between two concepts in C, i.e.:

$$k = \Re : C \times C \to C. \quad (42)$$

Definition 41 indicates that the compositional operations of concept algebra, $\bullet_c$, provide a set of coherent mathematical means and rules for knowledge manipulation. Because the relations between concepts are transitive, the generic topology of knowledge is a hierarchical network as shown in Figures 2 and 3.

Theorem 4. *The generic topology of abstract knowledge systems K is a hierarchical concept network.*

Theorem 4 can be proved by the nine compositional rules in concept algebra, particularly the composition/decomposition and aggregation/specification operations, as defined in the previous section.

Corollary 2. *The property of the hierarchical knowledge architecture K in the form of concept networks is as follows:*

a. **Dynamic:** *The knowledge network may be updated dynamically along with information acquisition and learning without destructing the existing concept nodes and relational links.*
b. **Evolvable:** *The knowledge network may grow adaptively without changing the overall and existing structure of the hierarchical network.*

The Hierarchical Model of Concept Networks

A concept network as a generic knowledge model has been widely studied in linguistics, computing, and cognitive informatics. The notion of the *semantic network* model for knowledge representation is first proposed by Quillian in 1968 (Matlin, 1998; Quillian, 1968; Reisberg, 2001), where the semantic memory is perceived as information represented in network structures with conceptual nodes and interrelations. The meaning of a given concept depends on other concepts to which it is connected in the network. The semantic network has been extended by a number of theories such as the *hypothetical network* (Colins & Loftus, 1975), the *adaptive control of thought - star* (ADT*) model (Anderson, 1983, 1991). The latter proposes that all cognition processes in thought are controlled by unitary network models.

This subsection develops the concept network model based on concept algebra and Wang's OAR model (Wang, 2007c; Wang & Wang, 2006) for knowledge representation, which treat a concept as a basic and adaptive unit for knowledge representation and thinking.

Definition 42. *A concept network CN is a hierarchical network of concepts interlinked by the set of nine composing rules $\Re$ concept algebra, i.e.:*

$$CN = \Re : \mathop{\times}_{i=1}^{n} C_i \to \mathop{\times}_{j=1}^{n} C_j \quad (43)$$

Theorem 5. *In a concept network CN, the abstract levels of concepts ℓ_c form a partial order of a series of superconcepts, i.e.:*

$$\ell_c = (\varnothing \preccurlyeq c_1 \preccurlyeq c_2 \preccurlyeq \ldots \preccurlyeq c_n \preccurlyeq \ldots \preccurlyeq \Omega) \quad (44)$$

where $\varnothing$ is the empty concept $\varnothing = (\perp, \perp)$, and Ω the universal concept, $\Omega = (\mathcal{O}, \mathcal{A})$.

According to Theorem 5 and Definition 42, a hierarchical structure of concepts in a given semantic environment Θ can be formally described by concept algebra. The algebraic relations and compositional operations of concept algebra enable the construction of hierarchical concept networks in a dynamic process.

Example 4. *A concrete concept network, pen and printer, can be illustrated in Figure 4. The concept network may be dynamically extended along with the development of related knowledge such as to*

Box 16.

$$
\begin{aligned}
&c_1(pen) = c_1(O_1, A_1, R^c{}_1, R^i{}_1, R^o{}_1) \\
&\text{where } O_1 = \{o_{11}, o_{12}, o_{13}\} = \{ballpoinbt, fountain, brush\}; \\
&\quad A_1 = \{a_1, a_2, a_3\} = \{a\_writing\_tool, using\_ink, having\_a\_nib\}; \\
&\quad A^*{}_1 = \{a_1, a_2, a_3, a_4\} = A \cup \{with\_an\_ink\_container\}; \\
&\quad R^c{}_1 = O_1 \times A_1 = \{(o_{11}, a_1), (o_{11}, a_2), (o_{11}, a_3)\} \cup \\
&\qquad \{(o_{12}, a_1), (o_{12}, a_2), (o_{12}, a_3)\} \cup \{(o_{13}, a_1), (o_{13}, a_2), (o_{13}, a_3)\}
\end{aligned} \tag{45}
$$

extend it to a more abstract concept network of stationery. In Figure 4, concept c_1(pen) may be formally described in concept algebra, see Box 16.

It is noteworthy that, according to Definition 10, the intension of c_1(*pen*) does not include the attribute a_4, because it is not commonly shared by all objects of the given concept. However, A_1* does include a_4 in the closure of attributes of the given concept *pen*.

Example 5. *An abstract concept network that is formed by the composition and aggregation of a set of related concepts c_0 through c_9, as well as objects o_1 through o_3, can be expressed in Figure 5.*

A formal description corresponding to the above concept network can be carried out using concept algebra as given below:

Figure 4. A concrete concept network

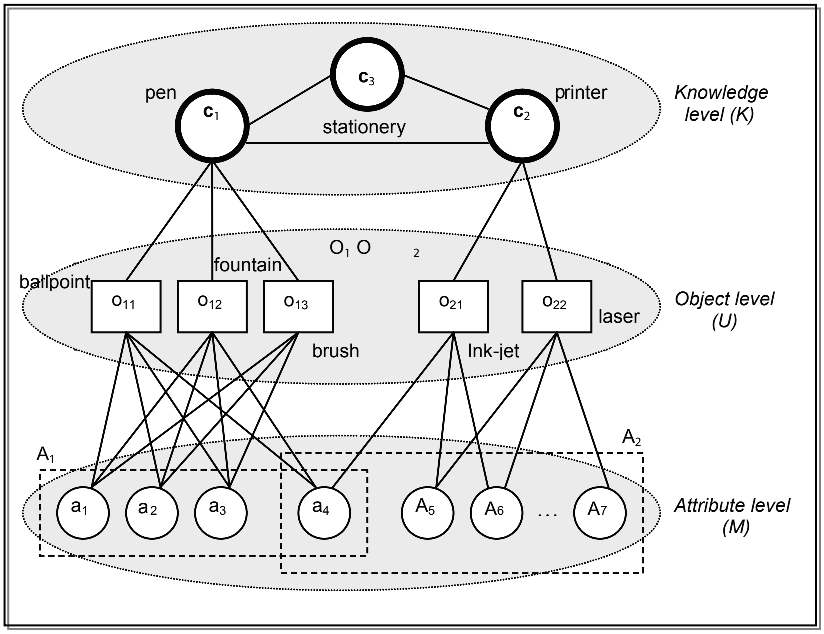
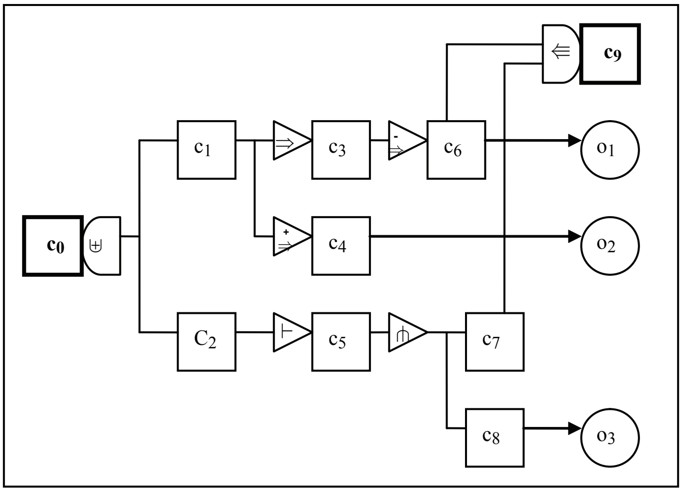


Figure 5. An abstract concept network



$$c_0 \uplus ((c_1 \Rightarrow c_3 \overset{-}{\Rightarrow} c_6 \mapsto o_1) \parallel (c_1 \overset{+}{\Rightarrow} c_4 \mapsto o_2)) \\ \parallel (c_2 \vdash c_5 \pitchfork (c_7 \parallel (c_8 \mapsto o_3))$$

$$c_9 \Lleftarrow (c_6 \parallel c_7) \tag{46}$$

The case studies in Examples 4 and 5 demonstrate that concept algebra and concept network are a generic and formal knowledge manipulation means that are capable to deal with complicated abstract or concrete knowledge structures and their algebraic operations. Further, detailed concept operations of concept algebra may be extended into a set of inference processes, which can be formally described by RTPA (Wang, 2002b, 2003, 2006a, 2007b, 2007d, 2008a, 2008d) as a set of behavioral processes.

CONCLUSION

A new mathematical structure, known as concept algebra, has been presented for abstract concepts and knowledge representation and manipulation. Concepts have been treated as the basic unit of cognition that carries certain meanings in almost all cognitive processes such as thinking, learning, reasoning, and system design. Abstract concepts have been formally modeled as a dynamic mathematical structure with internal attributes, objects, and their relations that possess the adaptive capability to grow in concept networks. The formal methodology for manipulating knowledge has been developed using concept algebra and concept networks. A number of case studies have been used to demonstrate the expressive power of concept algebra in knowledge representation and manipulation. A wide range of applications of concept algebra has been identified for solving common problems in cognitive informatics, logic, linguistics, psychology, knowledge engineering, data mining, software engineering, and intelligence science. One of the important applications of concept algebra is the formalization of object-oriented methodologies and the development of a rigorous semantics of UML as an industrial OO design languages. Autonomic machine learning and searching engines can be developed on the basis of concept algebra.

ACKNOWLEDGMENT

The author would like to acknowledge the Natural Science and Engineering Council of Canada

(NSERC) for its partial support of this work. The author would like to thank anonymous reviewers for their valuable comments and suggestions. The author is grateful to Dr. Y. Y. Yao for many insightful discussions.

REFERENCES

Anderson, J. R. (1983). *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Anderson, J. R. (1991). Is human cognition adaptive? *Behavioral and Brain Science, 14,* 71-517.

Chen, Y., & Yao, Y. Y. (2005). Formal concept analysis based on hierarchical class analysis. In *Proceedings of the Fourth IEEE International Conference on Cognitive Informatics (ICCI'05)* (pp. 285-292). Irvin, CA: IEEE CS Press.

Codin, R., Missaoui, R., & Alaoui, H. (1995). Incremental concept formation algorithms based on Galois (concept) lattices. *Computational Intelligence, 11*(2), 246-267.

Colins, A. M., & Loftus, E. F. (1975). A Spreading-activation theory of semantic memory. *Psychological Review, 82,* 407-428.

Ganter, B., & Wille, R. (1999). *Formal concept analysis.* Berlin, Germany: Springer.

Hampton, J. A. (1997). *Psychological representation of concepts of memory* (pp. 81-11). Hove, England: Psychology Press

Hurley, P. J. (1997), *A concise introduction to logic* (6th ed.). Belmony, CA: Wadsworth.

Matlin, M. W. (1998). *Cognition* (4th ed.). New York: Harcourt Brace.

Medin, D. L., & Shoben, E. J. (1988). Context and structure in conceptual combination. *Cognitive Psychology, 20,* 158-190.

Murphy, G. L. (1993). Theories and concept formation. In I. V. Mechelen et al. (Eds.), *Categories and concepts, theoretical views and inductive data analysis* (pp. 173-200). New York: Academic Press.

Quillian, M. R. (1968). Semantic memory. In M. Minsky (Ed.), *Semantic information processing*. Cambridge, MA: MIT Press.

Reisberg, D. (2001). *Cognition: Exploring the science of the mind* (2nd ed.). New York: Norton.

Smith, E. E., & Medin, D. L. (1981). *Categories and concepts.* Cambridge, MA: Harvard University Press.

Wang, Y. (2002a). Keynote lecture: On cognitive informatics. In *Proceedings of the First IEEE International Conference on Cognitive Informatics (ICCI'02)* (pp. 34-42). Calgary, Canada: IEEE CS Press.

Wang, Y. (2002b). The real-time process algebra (RTPA). *The International Journal of Annals of Software Engineering, 14,* 235-274.

Wang, Y. (2003). Using process algebra to describe human and software system behaviors. *Brain and Mind*, *4*(2), 199-213.

Wang, Y. (2006a, July). Cognitive informatics and contemporary mathematics for knowledge representation and manipulation, invited plenary talk. In *Proceedings of the First International Conference on Rough Set and Knowledge Technology (RSKT'06)* (LNAI 4062, pp. 69-78). Chongqing, China: Springer.

Wang, Y. (2006b). On abstract systems and system algebra. In *Proceedinngs of the Fifth IEEE International Conference on Cognitive Informatics (ICCI'06)* (pp. 332-343). Beijing, China: IEEE CS Press.

Wang, Y. (2006c). On concept algebra and knowledge representation. In *Proceedings of the Fifth IEEE International Conference on Cognitive Informatics (ICCI'06)* (pp. 320-331), Beijing, China: IEEE CS Press.

Wang, Y. (2006d, March). On the informatics laws and deductive semantics of software. *IEEE Transactions on Systems, Man, and Cybernetics (C), 36*(2), 161-171.

Wang, Y. (2007a). The cognitive processes of formal inferences. *The International Journal of Cognitive Informatics and Natural Intelligence, 1*(4), 75-86.

Wang, Y. (2007b). Keynote speech: On theoretical foundations of software engineering and denotational mathematics. In *Proceedings of the Fifth Asian Workshop on Foundations of Software* (pp. 99-102). Xiamen, China:.

Wang, Y. (2007c). The OAR model of neural informatics for internal knowledge representation in the brain. *The International Journal of Cognitive Informatics and Natural Intelligence, 1*(3), 64-75.

Wang, Y. (2007d, July). Software engineering foundations: A software science perspective. In *CRC Series in Software Engineering: Vol. 2*. CRC Press.

Wang, Y. (2007e). The theoretical framework of cognitive informatics. *The International Journal of Cognitive Informatics and Natural Intelligence, 1*(1), 1-27.

Wang, Y. (2008a, April). Deductive semantics of RTPA. *The International Journal of Cognitive Informatics and Natural Intelligence, 2*(2), 96-121.

Wang, Y. (2008b, April). On system algebra: A denotational mathematical structure for abstract system modeling. *The International Journal of Cognitive Informatics and Natural Intelligence, 2*(2), 20-43.

Wang, Y. (2008c). On the big-R notation for describing iterative and recursive behaviors. *The International Journal of Cognitive Informatics and Natural Intelligence, 2*(1), 17-18.

Wang, Y. (2008d, April). RTPA: A denotational mathematics for manipulating intelligent and computational behaviors. *The International Journal of Cognitive Informatics and Natural Intelligence, 2*(2), 44-62.

Wang, Y., & Wang, Y. (2006, March). On cognitive informatics models of the brain. *IEEE Transactions on Systems, Man, and Cybernetics (C), 36*(2), 203-207.

Wille, R. (1982). Restructuring lattice theory: An approach based on hierarchies of concepts. In I. Rival (Ed.), *Ordered sets* (pp. 445-470). Reidel, Dordrecht

Wilson, R. A., & Keil, F. C. (Eds.). (1999). *The MIT encyclopedia of the cognitive sciences*. Cambridge, MA: The MIT Press.

Yao, Y. Y. (2004). Concept formation and learning: A cognitive informatics perspective. In *Proceedings of the Third IEEE International Conference on Cognitive Informatics (ICCI'04)* (pp. 42-51). Victoria, British Columbia, Canada: IEEE CS Press.

Yingxu Wang is professor of cognitive informatics and software engineering, director of International Center for Cognitive Informatics (ICfCI), and director of Theoretical and Empirical Software Engineering Research Center (TESERC) at the University of Calgary. He received a PhD in software engineering from The Nottingham Trent University, UK, in 1997, and a BSc in electrical engineering from Shanghai Tiedao University in 1983. He was a visiting professor in the Computing Laboratory at Oxford University and Department of Computer Science at Stanford University during 1995 and 2008, respectively, and has been a full professor since 1994. He is the editor-in-chief of the International Journal of Cognitive Informatics and Natural Intelligence *(*IJCINI*), and editor-in-chief of the* CRC Book Series in Software Engineering*. He has published over 300 journal and conference papers and 11 books in software engineering and cognitive informatics, and won dozens of research achievement, best paper, and teaching awards in the last 28 years, particularly the IBC 21st Century Award for Achievement "in recognition of outstanding contribution in the field of Cognitive Informatics and Software Science," and the groundbreaking book on* Software Engineering Foundations: A Software Science Perspective.