Proceedings of ICAD2011
The Sixth International Conference on Axiomatic Design
Daejeon – March 30-31, 2011
ICAD-2011-01

A LOGIC-BASED FOUNDATION OF AXIOMATIC DESIGN

Stephen C.-Y. Lu
sclu@usc.edu
University of Southern California
Los Angeles, CA
USA

Ang Liu
angliu@usc.edu
University of Southern California
Los Angeles, CA
USA

ABSTRACT

One of the most essential and unique features of the Axiomatic Design Theory is its clear differentiation between the "what" and "how" decisions. This delineation sets the origin for realization and specialization procedures, and, along with the domain and layer concepts respectively, constitutes a unique two-dimensional design framework. Based on formal logic studies, this paper presents a theoretical underpinning to elucidate the fundamental reasons for delineating “what” from “how” decisions, hence providing guidance to justify and execute the mapping and decomposition operations prescribed by the Axiomatic Design Theory. This logic-based foundation also establishes a synthesis reasoning framework which can be seen as a theoretical generalization of the Axiomatic Design Theory to better support design synthesis.

Keywords: Axiomatic Design, synthesis, logic.

1 INTRODUCTION

Compared with other design approaches, one of the most essential and unique features of the Axiomatic Design (AD) theory is its clear differentiation between the "what" and "how" design decisions. However, due to the lack of a theoretical foundation, the important “what to how” mapping prescribed by the AD theory causes much confusion and many difficulties when applied to the design practice, especially when used as a synthesis reasoning tool. As a consequence of such difficulty, designers often fail to take advantage of the power of the AD theory during the synthesis phase of design to create new options. Instead, many designers limit the usage of the theory to the evaluation phase and the analysis of already generated options. Based on formal logic studies, this paper presents a theoretical underpinning to explain the fundamental reasons for clearly delineating “what” from “how” decisions, and provides justification and guidance to mapping and decomposition operations prescribed by the AD Theory.

In this paper, we first discuss the importance of searching for theoretical foundations to support the differentiation between “what” and “how” decisions. We then present some basic concepts from formal logic which are relevant to the two-dimensional decision framework of the AD theory. Next, we explain how these logic-based concepts can be used to distinguish and guide the mapping and decomposition operations in AD. We also expand this logic foundation to build a generic synthesis reasoning framework. This framework can be seen as a generalization and used as a complement of the AD theory in order to deepen its theoretical significance and broaden its practical impact in design. Finally, we summarize lessons learned and draw some conclusions to guide future research.

2 WHY DIFFERENTIATE BETWEEN “WHAT” AND “HOW” DECISIONS IN DESIGN?

Generally speaking, three different approaches have been developed in the engineering design research community to date: algorithm-based, decision-based, and axiom-based design. The algorithm-based approach [Pahl and Beitz, 1996] relies mostly on descriptive studies of engineering practices to structure design procedures and proscribe detailed steps for the designer to follow. This approach is easier to adapt in design practice, but lacks a theoretical basis for objective validations. The second type is exemplified by the Decision-Based Design (DBD) approach [Hazelrigg, 1998], which is derived directly from classical decision science and rational decision theory. Although this approach has sound theoretical foundations, it is often limited by its real world applicability. Lastly, the axiom-based approach, which is best represented by the Axiomatic Design Theory proposed by Nam P. Suh [Suh, 1990], tries to "strike a balance" between theory and practice by proposing a few “axioms” derived from good design practices and stating them as “fundamental truths” in building the design theory. Many further research and development efforts have been devoted to improve the applicability and effectiveness of the AD theory in engineering design [Nordlund *et al.*, 1996; Suh, 2001].

To put it concisely, the AD theory can be best summarized and understood by the following concepts:

1. A two-dimensional design framework that consists of the notion of “domains” to categorize different types of design decisions and “layers” to capture their different abstraction levels,
2. An iterative zigzagging design process that alternates between pairs of two adjacent domains while decomposing higher-level abstract decisions into lower-level detailed ones across layers, and
3. Two generic design axioms, namely the Independence Axiom and the Information Axiom, which guide the comparison and selection of good design decisions.

One of the most essential concepts of the AD theory is the two-dimensional decision framework. When using the theory, the designer can generate as many layers as practically allowed, but the "domain" is limited to only four types: (1) the Customer Need (CN) domain, (2) the Functional Requirement (FR) domain, (3) the Design Parameter (DP) domain, and (4) the Process Variable (PV) domain. The zigzagging design process consists of repeatedly making decisions across domains from upstream CNs to downstream PVs, and, at the same time, making decisions across layers from higher abstract to lower detail levels. At each decision point during this zigzagging process, the Independence Axiom is used to guide the creation and characterization of multiple design concepts (decision alternatives or options) into three categories: uncoupled, decoupled, and coupled; and then the Information Axiom is employed to compare all created design options to select the "least risky" concept (in terms of its possible physical implementation) as the final design decision, which will be carried onto the next decision point in the process.

When comparing the AD theory with other design theories and approaches, it is clear that one of the most unique (as well as important and essential) features of AD is its requirement on clearly distinguishing design decisions into two different kinds with regards to the notions of domain and layer (hence the 2-D design framework). The former is called a "mapping" operation from "what" to "how" across two neighboring domains, and the latter is called a "decomposition" operation from "what to what" (or "how to how") across two adjacent layers. Because there are four distinctive design domains, the designer must create four separate decision hierarchies simultaneously when using the AD theory in design practice. This 2-D "mapping-decomposition" (or "what-how") framework is in sharp contrast with other approaches, such as the analytical hierarchical process (AHP) [Saaty, 1990], which focuses on decomposing an abstract decision at higher layer repeatedly into more detailed layers along the same direction to create a single hierarchy. Table 1 compares the key differences between the traditional AHP procedure and the AD theory.

Table 1. Comparison between the AHP and AD theory.

	AHP Process	AD Theory
Decision Framework	Repeated one-dimensional decomposition from "abstract to detail"	Repeated 2-D mapping from "what to how" and decomposition from "abstract to detail"
Decision Process	Fish-bone like decision-tree diagram (leads to a single hierarchy)	Alternate zigzagging between mapping and decomposition (leads to four separate hierarchies)
Decision Selection	Traverse all decision links to aggregate those subjectively-assigned influencing factors and weights	Axiom 1 to create and characterize multiple options, and then Axiom 2 to rank-order and select the lowest risk one

Before presenting our research, let's further explain the theoretical significance and important impact of the answer (or the lack thereof) to this question on design research and practice at large. Since its inception, the AD theory has received much criticism and faced many challenges from the research community. From a theoretical point of view, one of the root sources for these disagreements lies in the nature of what is (or can be) counted as an "axiom". An axiom, according to its dictionary definition, is a fundamental statement which must be accepted as true but cannot be derived from other known theories or accepted laws. Although the history of science reveals that axioms have played a critical role in the pursuit of many scientific studies, those who always demand analytical "proof" of everything are not comfortable nor satisfied unless some theoretical foundations can be provided to reasonably explain or logically derive the statement. Otherwise, they would disprove the proposed axiom and reject all of its derivatives as more-or-less a subjective and religious belief. Whether these design axioms are objective (which needs some theoretical backing) or subjective (which could be biased by individuals' experiences) is, in fact, at the center of the research debates surrounding the AD theory.

From a practical perspective, the lack of a theoretical underpinning for the design axioms hinders the effective teaching, systematic learning, and appropriate use of the AD theory in engineering practice. For example, while it is easy to explain and illustrate the difference between a "what" and a "how" decision in the classroom, the practitioner is often easily confused by the two when using the AD theory in real world applications. The confusion becomes worse because an upstream "how" must also be viewed as a downstream "what" at the same time. The designers are often trapped by the bewilderment between mapping and decomposition operations, always having difficulties in carrying out the zigzagging procedure systematically and resulting in bad mixes of "what" and "how" in their design hierarchies. Such a difficulty leads to the fact that the designers often fail to take advantage of the real power of the AD theory during the synthesis phase of design. For example, the Independence Axiom, which can/should be used to synthetically guide the creation of design alternatives that are functionally independent of each other during the synthesis phase, is often used merely as a tool to analytically represent and compare the types of dependency among multiple design options (by the shape of the design matrix) during the alternative evaluation phase. This is evident by the fact that the majority of reported AD applications to date [Gebala and Suh, 1999; Kulak and Kahraman 2005; Suh, 2001a] have been focused on the usage of the two axioms to analyze, evaluate and compare feasible design options, rather than to synthesize and generate new design ideas.

In order to settle the debate on whether the design axioms should be universally accepted as objective rules or simply treated as subjective guidance, we should first examine if there is a theoretical foundation which can explain the reason for clearly delineating a "what" from a "how" decision; and, if so, how to use this foundation to guide mapping and decomposition operations differently. Because design is intrinsically a human activity and good designs are often the result of systematic reasoning, this foundation is most likely to be found in disciplines that study the fundamentals of human reasoning. One obvious candidate is formal logic, which is the basis for investigating human cognition and reasoning. The rest of this paper describes our research efforts in finding such a theoretical foundation for the AD theory from formal logic studies.

It is important to point out that something (such as a theory) which has a logic-based explanation does not mean that it, by itself, is logical in the strict logic sense. In other words, by adapting a logic-based foundation by no means suggests that we intend to develop the AD theory as a "logical" design theory. In fact, due to the socio-technical nature of engineering design problems, we take the stand that a useful design theory can never be strictly based on pure logic; but rather it has to be rationally (rather than logically) formulated to align with the characteristics of human cognition [Lu, 2009]. This is different from other previous research efforts that attempt to approaching design decisions based on formal logic [Zeng, 2002].

3 RELEVANT EPISTEMOLOGICAL CONCEPTS FROM FORMAL LOGIC STUDIES

Generally speaking, to design something is to "synthesize" purposefully from a set of relatively abstract requirements and constraints to generate some tangible plans and concrete specifications. During this creation process, the designer must perform synthesis reasoning which uses abductive logic and domain knowledge to make various propositions that transform the state of the design from abstract to detailed. This type of "propositional knowledge" plays an important role in design synthesis reasoning, and its scope and nature have been a major focus in the field of epistemology. Modern epistemological studies have clearly defined two different forms of propositional knowledge, namely "know-that" and "know-how". In mathematics, for example, 2 + 2 = 4 can be either a "knowing-that" knowledge, which merely states the fact that the sum of 2 and 2 is 4, or a "know-how" knowledge which implies knowing how to add any two numbers. That is to say that, if 2 + 2 = 4 is proposed only as a "knowing-that" knowledge; then nothing is said about 2 + 3 with this particular proposition. Hence, 2 + 3 = 5 must be affirmed by another separate "knowing-that" proposition. On the other hand, if 2 + 2 = 4 is proposed as a "knowing-how" knowledge, then the same proposition can also lead to the inference of knowing 2 + 3 = 5, or adding any two numbers together, for that fact. Such epistemological difference can also be illustrated by the example of the act of balance involved in riding a bicycle. The theoretical knowledge involved in maintaining a state of balance in physics (i.e., "knowing-that" knowledge) cannot substitute for the practical knowledge of how to ride a bike (i.e., "knowing-how" knowledge).

Both "knowing-that" and "knowing-how" propositional knowledge are needed for synthesis reasoning in design. However, they should be used differently and this will lead to different dependency relationships in the final design hierarchy. This is because, from the formal logic point of view, "knowing-that" and "knowing-how" knowledge play different roles in making propositions. In general terms, we can say that the "knowing-that" knowledge affirms the facts, whereas the "knowledge-how" knowledge asserts the methods (or reasons) behind the facts. The logician Immanuel Kant [Kant, 1781] used the terms "analytic" and "synthetic" to divide propositions into two types. He defines an "analytic proposition" as a proposition type whose predicate concept is "contained in" its subject concept. For examples, "bachelors are unmarried", "triangles have three sides", and "forces have equal reacting forces" are all analytic propositions; because their predicate concepts (i.e., unmarried, three sides, reacting forces) are all contained within the definitions of the subject concepts (i.e., bachelors, triangles, forces). Analytic propositions use the "knowing-that" knowledge to deductively affirm predicates, whose definitions are fully contained within that of the subject. They establish the "part-of" relationships within a single design hierarchy.

In contrast, a "synthetic proposition" is defined as a proposition whose predicate concept is "not contained in" its subject concept. For examples, "bachelors are happy", "creatures with hearts have kidneys", and "powers are generated by engines" are all synthetic propositions; because their subject concepts (i.e., "bachelors", "creatures with hearts" and "powers") "do not" necessarily contain their predicate concepts (i.e., "happy", "have kidneys" and "engines"). In other words, the dependency relationships created between the subject and the predicate via synthetic propositions is NOT the "part-of" type as with the case of analytic propositions. This is a very important difference that must be well understood because it provides the logic foundation upon which the two-dimensional decision framework of the AD theory is developed. As will be explained next, in the context of design, synthetic propositions employ the "knowing-how" knowledge to abductively establish the "means-of" dependency relationships across multiple hierarchies (rather than a single hierarchy), and are the basis of the "mapping" operation in the AD theory.

4 MULTIPLE HIERARCHIES TO DIFFERENTIATE ANALYTIC FROM SYNTHETIC PROPOSITIONS

The previous section has established the epistemological foundation from formal logic studies that elucidates the important dissimilarity between analytic propositions and synthetic propositions. When the designer makes these two different propositions repeatedly, various dependency relationships are created. In a typical design, these relationships can become very complex and must be structured properly in order to support reasoning. Based on the centennial work by Herbert Simon [1996] that proposed "hierarchy" as a good structure to "mask" system complexity, entities of a system are often organized as a hierarchy in order to take advantages of information hiding and property inheritance, among other benefits.

In general, a hierarchy is a structure that directly or indirectly links entities in either vertical or horizontal directions to capture dependency relationships among entities (e.g., propositions). Strictly speaking, the only direct relationships in a hierarchy are to one's immediate superior or subordinates. Table 2 summarizes the similarities and

Table 2. Similarities and differences among the hierarchies generated in different disciplines.

	Methodology	Level of	Types of Entities	Dependency
Product Design	Axiomatic Design	Abstraction	WHAT vs. HOW	"a means of"
Software Architecting	Object-oriented programming	Implementation	Class vs. Subclass	"a kind of"
Organizational Structure	Hierarchical organization	Power /Authority	Superior vs. Subordinate	"a subordinate of"
Control System	Hierarchical control system	Planning and execution time	Superior vs. Subordinate nodes	"a task of"
Functional Modeling	IDEF0	Data flow	Input vs. Output	"a function of "

differences among diverse hierarchies generated for different disciplines (i.e., product design, organization, software architecting, control system, functional modeling).

Synthesis reasoning in the context of design results in an implementable "means/how" (i.e., predicate) design hierarchy that can fully satisfy the intended "ends/what" (i.e., subject). On one hand, the concept of to-be-created "means/how" is not "part-of" the concept of intended "ends/what"; hence, synthetic propositions must be performed to establish the NOT "part-of" dependency relationship between predicate and subject. On the other hand, both the "means/how" and "ends/what" must form a separate hierarchy each with multiple layers to make the initial intent fully understood and the final solution consistently implementable; hence, analytical propositions are needed to establish the "part-of" dependency relationships between superior and subordinate entities. This leads to the fact that all predicate entities derived from synthetic propositions are of different kinds (and should belong to different families) and therefore must be organized into a separate hierarchy. Whereas, the subordinate entities created by analytic propositions are of the same kind (i.e., either ends or means) and hence can be organized into the same hierarchy. Figure 1 illustrates these two different kinds of propositions and the different dependency relationships among them in a multi-hierarchy structure.

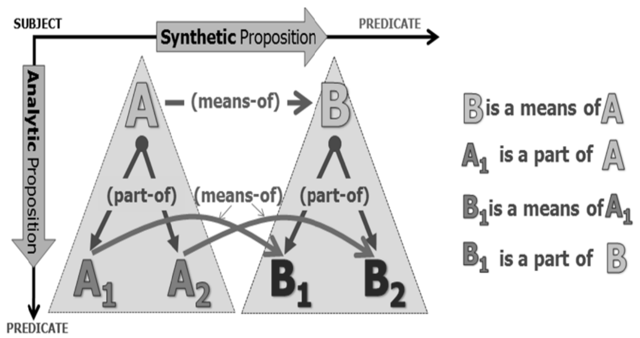

Figure 1. Dependency relationships and hierarchies for analytic and synthetic propositions.

For example, when an analytic proposition is affirmed to a general subject A during synthesis reasoning, it results in two more specific subordinate entities A_1 and A_2 (Figure 1). Because the analytic proposition uses the "knowing-that" knowledge to affirm predicates that are contained within the definition of the subject (i.e., they are within the same family), it establishes the "part-of" dependency relationships between A and A_1 (and between A and A_2). This means that both A_1 and A_2 are "part-of" A. Other than this, no other explicit inference (hence no direct dependency relationship) is made directly between A_1 and A_2 via such an analytic proposition. Within this single hierarchy, A is the direct superior of A_1 and A_2 by, and only by, vertical links. In object-oriented programming [Rumbaugh *et al.,* 1991], this dependency is similar to the relationship between a class (A) and its objects (A_1 and A_2). Property inheritance and information encapsulation are made possible vertically through the "part-of" dependency relationships within a single hierarchy by OO programming. As indicated in Figure 1, all element (or children) predicates derived from the whole (or parent) subject (A) share its common properties, and therefore are placed within the same single hierarchy. This is also to say that those subjects or predicates which do not completely share A's properties, such as B, B_1 and B_2 in Figure 1, must be organized separately into a different hierarchy among themselves (who share same properties).

On the other hand, if a synthetic proposition is abductively made on the same general subject A by the designer, it will result in a very different type of dependency relationship. This is because that, according to the definition of synthetic proposition, the asserted predicate (say B in Figure 1) is NOT contained within the definition of the subject A. Therefore, we cannot infer that "B is a part-of A", as with the case in analytic propositions. In other words, although A_1 and B are both predicates derived from the same subject A, because of different types of propositions made, they are of dissimilar kinds and have very different relationships with A. Property inheritance and information encapsulation that exist between A and A_1 (and A_2) do not hold true for the relationship between A and B, B_1 and B_2; therefore, B (and B_1 and B_2) cannot be placed in the same hierarchy as A (and A_1 and A_2). Instead, they must be organized within a separate hierarchy during synthesis reasoning.

5 A LOGIC FOUNDATION FOR DECOMPOSITION AND MAPPING OPERATIONS PRESCRIBED IN THE AXIOMATIC DESIGN THEORY

The previous section has explained the difference between analytic and synthetic propositions as well as the necessity for creating a separate (i.e., an additional) hierarchy to organize synthetic propositions during synthesis reasoning.

Table 3. Comparison between different propositions, knowledge, relationships and structures.

	Kinds of Proposition	Types of Knowledge	Nature of Relationship	Synthesis Operation	Reasoning Direction	Hierarchical Structure
Mapping Across Domains	Synthetic	Knowing-how	Means-of	Realization	Horizontal	Across two hierarchies
Decomposition Across Layers	Analytic	Knowing-that	Part-of	Specialization	Vertical	Within a single hierarchy

Based on this theoretical background, we can now use these basic concepts to justify and guide the mapping and decomposition operations that underline the domain-vs.-layer 2-D reasoning framework upon which the AD theory was developed.

Since the purpose of design synthesis is to create some concrete "means" to achieve the intended "ends", we can define the logic association established by synthetic propositions in design as the "means-of" dependency relationship between the subject and its predicates. In other words, we can say that the predicate B is a "means-of" the subject A; or A is "realized-by" B via a synthetic proposition. The resulting predicate B is not the same kind as the subject A in this case. Therefore, when B is to be further specified by analytic propositions, they should be placed into a different hierarchy than that for the A's as indicated in Figure 1. Unlike the "part-of" relationships which exist "vertically" within a hierarchy, we can describe the "means-of" relationships as moving "horizontally" across adjacent hierarchies.

In short, analytic propositions in synthesis reasoning use the knowing-that knowledge to establish the "part-of" dependency relationships vertically between the subject and its predicates with property inheritance and information encapsulation within a single hierarchy. Synthetic propositions, on the other hand, use the knowing-how knowledge to create the "means-of" dependency relationships between the subject and the predicates without property inheritance and information encapsulation across two hierarchies. We call the horizontal assertion of the "means-of" dependency relationships as "realization" and the vertical declaration of the "part-of" dependency relationships as "specialization" to support synthesis reasoning in tandem.

Recall that the key feature of the AD theory is to organize different kinds of design decisions into four domains (e.g., CN, FR, DP and PV) within four separate hierarchies. Compared with the traditional AHP process that only vertically decomposes an intangible subject into more tangible predicates within a single hierarchy, the AD theory suggests an additional reasoning operation, called mapping, which makes propositions across two adjacent design domains. Based on the above explanations, it is clear that the mapping operation in AD should be based on the synthetic proposition, when the designer should reason horizontally across two different hierarchies using the know-how knowledge; whereas the decomposition operation in AD is mostly based on the analytic proposition, when the designer must reason vertically within one hierarchy using the know-that knowledge. Therefore, when using the AD theory to perform synthesis reasoning in design, the designer can rely on the epistemological difference between know-how and know-that knowledge to clearly differentiate and systematically guide the synthetic and analytic propositions during mapping and decomposition operations accordingly.

Table 3 recaps how the relevant scientific underpinnings and concepts explained in Sections 3 and 4 correlate to the mapping and decomposition operations in the AD theory. Equipped with these basic concepts and logic foundations in this table, the designer will be more able to use the AD theory's 2-D decision framework effectively to synthetically generate new design options (as oppose to merely analyzing/comparing multiple already created design alternatives) via the zigzagging design procedure with alternate uses of mapping and decomposition operations. This not only provides a theoretical foundation for the AD theory in research, but also overcomes the difficulty of using the theory to creatively perform design synthesis in practice.

6 SYNTHESIS REASONING FRAMEWORK AS A GENERALIZATION OF THE AXIOMATIC DESIGN THEORY

Building upon the above logic-based theoretical foundation, we can further develop a generic synthesis reasoning framework which can be seen as a theoretical generalization and used as a complement of the AD theory to effectively support design synthesis [Lu and Liu, 2011].

In general, synthesis reasoning in formal logic represents a rational "leap of faith" from a relatively intangible subject ($P_{i,j}$) to a more tangible predicate ($P_{i+1,j+1}$) by making a series of abductive propositions, where i and j denote the synthetic and analytic propositions, respectively. Based on this logic foundation, design synthesis can be modeled as a repeated abduction process from an abstract intent (i.e., what) to a concrete instantiation (i.e., how). Figure 2 below illustrates a typical synthesis reasoning process in design from $P_{i,j}$ to $P_{i+1,j+1}$. The process consists of two sequential stages: the alternative creation stage and the alternative selection stage. The goal of alternative creation is to ideate a few qualified instantiations for further comparison. Whereas, the goal of alternative selection is to choose a unique instantiation as the final outcome (i.e., $P_{i+1,j+1}$) of design synthesis.

In the alternative creation stage, given $P_{i\ j}$, the designer must first mentally form a "nucleus" in order to focus his/her creative attentions. In other words, starting from "all things are possible" initially (i.e., the solution-free thinking desired by innovative design should begin with all possible alternatives without any limitation), a bounded small "space for consideration" (to which $P_{i+1,j+1}$ must belong) must be established carefully first.

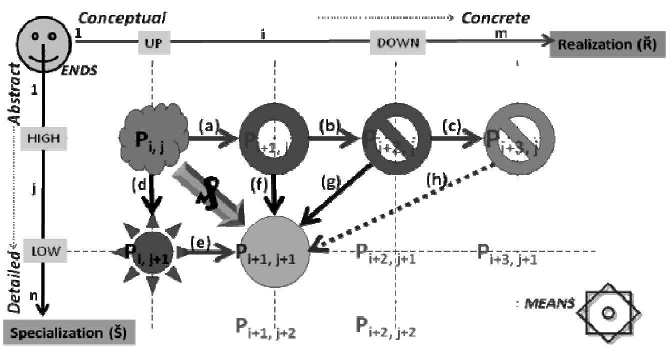

Figure 2. Typical synthesis reasoning from $P_{i,j}$ to $P_{i+1,j+1}$.[Lu and Liu, 2011]

Three different synthesis reasoning operations are defined and applied in tandem here. The first is the "Realization Operation (Ř)" that uses synthetic proposition to create the "means-of" relationship between $P_{i,j}$ and $P_{i+1,j}$. In other words, the designer must think horizontally along the same level of abstraction and ask "what are the possible $P_{i+1,j}$ that could be the means of realizing $P_{i,j}$?". Ř operation can be seen as the generalization of the horizontal "mapping" operation in the AD theory. The second is the "Specialization Operation (Š)" that uses analytic proposition to create the "part-of" relationship between $P_{i,j}$ and $P_{i,j+1}$. In other words, the designer must think vertically within the same decision domain and ask "what are the possible $P_{i,j+1}$ that could be a part of $P_{i,j}$?" The Š operation can be regarded as the generalization of the vertical "decomposition" operation in the AD theory. The third is the "Bounding Operation (□)" which assures that the resulting $P_{i+1,j+1}$ are limited by both domain-independent axioms as well as domain-dependent constraints. In other words, the designer must also think diagonally across one domain and one layer, and ask "what are the possible $P_{i+1,j+1}$ that would be within the boundary of limits imposed by these axioms and constraints?" In short, during the alternative creation stage, the final limited "space for consideration" ($P_{i+1,j+1}$) is formed by simultaneously considering the intersection between $P_{i+1,j}$ and $P_{i,j+1}$ that also meets some domain-independent as well as domain-dependent constraints.

The domain-independent constraints that must be included via the □ operation include the three criteria of the Independence Axiom from the AD theory: i.e., complete, minimal and independence. That is to say that, those alternatives within the limited "space for consideration" must completely satisfy the design intent expressed by $P_{i,j}$, without any redundancy (or duplication) among themselves, and be functionally independent from each other.. The domain-dependent constraints that must be considered via the □ operation have two kinds. When i=1 and j=1 for $P_{i+1,j}$ and $P_{i,j+1}$, the constraints are those design restrictions imposed onto the designer by corporations, policies, regulations, and markets as well as the known resources (such as time, budget, etc.) limits. For other instances (i.e., i>1, j>1), the constraints are those propositions that have been made previously at the upper abstraction layer and the downstream domain.

The combined considerations among the above "means-of", "part-of", and "constraint-by" (with both domain-independent axioms and domain-dependent constraints) operations will lead to a small limited space for consideration that consists of "a few high quality alternatives" at the conclusion of the alternative creation stage of design synthesis reasoning. These few alternatives will then become the candidates of comparison and choice during the alternative selection stage next.

To arrive at a unique $P_{i+1,j+1}$, certain selection methods must be introduced at the selection stage of design synthesis reasoning. In general, alternative selection always involves some sorts of comparison (i.e., evaluation and ranking) of alternatives based on their relevant merits (or estimated consequences) that are of interest to the designers. In later design stages, the merits are mostly derived from the technical performances of each alterative based on the objective brute reality knowledge of the application domain. However, during the early design stages when the decisions are more abstract without specific design parameters, such brute reality knowledge and objective evaluation models are often neither available nor possible. Instead, subjective human preferences driven by competing social realities are often the true driving force (and often the only possibility) behind the comparison and selection of the most agreeable alternative at the early design stages. Unfortunately, once subjectivity enters the decision making process and involves multiple designers, each with different preferences, the alternative selection task becomes very complicated.

The proposed synthesis reasoning framework utilizes two domain-independent methods in tandem to compare and select candidate alternatives. For early-stage design decisions, a specific preference-aggregation method guides designers to go through 3 sequential stages (i.e., preference formation, preference evaluation, and preference aggregation) to combine multiple individual preferences into a single team preference [Lu and Liu, 2011]. For later-stage design decisions, the Information Axiom from the AD theory is employed to rank-order candidate alternatives.

The proposed synthesis reasoning framework can be seen as a generalization and used as a complement of the AD theory, because (see Table 4 on next page):

1. It is also a 2-D decision framework that uses a horizontal "conceptual-concrete" spectrum and a vertical "abstract-detail" spectrum to represent the synthetic propositions and analytic propositions respectively.
2. The Ř and Š operations in the synthesis reasoning framework can be regarded as the generalization of the "mapping" operation and the "decomposition" operation in the AD theory respectively.
3. The framework defines an additional □ operation to manage domain-independent axioms and domain-dependent constraints included in the AD theory. On one hand, the "constrained-by" relationship via the □ operation ensures that the Independence Axiom must be utilized as the alternative creation principal as opposed to the alternative selection criteria in synthesis reasoning. On the other hand, the "constrained-by" dependency on domain-dependent constraints more explicitly indicates

how the zigzagging design process is carried out in design decision making practice.

4. The decision process of this synthesis reasoning framework also follows a zigzagging design process as the AD theory, by applying the three reasoning operations (i.e., Ř, Š, and □) in a specific sequence.
5. Two generic design axioms prescribed by the AD theory are both adopted as objective decision rules in this framework. The Independence Axiom is utilized as a domain-independent constraint during alternative creation stage to synthetically create instantiations that are functionally independent of each other. As well, the Information Axiom is used as a domain-independent selection criteria to compare and rank-order candidate instantiations.

Table 4. Comparison between the synthesis reasoning framework and the Axiomatic Design theory.

	Synthesis Reasoning Framework	The AD Theory
Basis of Theory	Logic-based	Axiom-based
Classification of Constraint	Yes	No
Classification of Reasoning Operation	Yes	Yes
Means-of Dependency	Realization	Mapping
Part-of Dependency	Specialization	Decomposition
Constrained-by Dependency	Bounding	No
Decision Framework	2-Dimensional	2-Dimensional
Horizontal	Conceptual-Concrete Spectrum	Four Domains
Vertical	Abstract-Detail Spectrum	Multiple Layers
Decision Process	Zigzagging	Zigzagging
Decision Selection	Yes	Yes
Merit of Comparison	Subjective preference and objective criteria	Objective criteria
Selection Method	Preference Aggregation; Information Axiom	Independence Axiom; Information Axiom

7 SUMMARY AND CONCLUSIONS

1. This paper attempts to find a theoretical foundation that can explain the fundamental reasons for clearly delineating the "what" from "how" decision in the AD theory. The answer to this question helps to settle the debate whether the design axioms (i.e., Independence Axiom and Information Axiom) prescribed by the AD theory should be commonly accepted as objective decision rules or simply treated as subjective guidance.
2. Based on relevant theory from formal logic, this paper builds a theoretical foundation that clearly distinguishes the essential difference between analytic propositions (made by "know-what" knowledge) and synthetic propositions (made by "know-how" knowledge), and the necessity for creating a multi-hierarchy framework to organize synthetic propositions. This logic-based theoretical foundation can be used to justify and guide the decomposition and mapping operations that underlines the domain-vs.-layer 2-dimensional decision framework upon which the AD theory was developed.
3. Built upon this logic foundation, a synthesis reasoning framework (including reasoning operations, decision process, and selection methods) is developed. The new framework can be seen as a generalization and used as a complement of the AD theory to enhance its more effective applications as a synthesis (instead of analysis) decision framework.
4. The future work of this research includes deriving some specific theorems of abductively making propositions (i.e., synthetic proposition and analytic proposition) in synthesis reasoning, that are compatible with relevant operations (i.e., mapping and decomposition) in the AD theory. Some design experiments are being conducted to test the performance of the proposed synthesis reasoning framework.

8 ACKNOWLEDGEMENTS

We acknowledge the continuous support, encouragement and inspiration from Professor Suh Nam Pyo in our long-term research pursuits relating to the Axiomatic Design Theory and innovative design thinking.

9 REFERENCES

[1] Gebala, D.A., Suh, N.P., "An Application of Axiomatic Design," *Journal of Mechanical Design*, Vol. 121, No. 3, pp. 342-347, 1999.

[2] Hazelrigg, G.A., "A Framework for Decision-Based Engineering Design", *ASME Journal of Mechanical Design*, Vol. 120 , 653-658, 1998.

[3] Kant, I., *Critique of Pure Reason*, Wood Cambridge University Press, 1781. ISBN: 0-521-65729-6.

[4] Kulak, O., and Kahraman, C., "Fuzzy Multi-Attribute Selection among Transportation Companies Using Axiomatic Design and Analytic Hierarchy Process", *Information Sciences*, Vol. 170, 2005, pp. 191-210, 2005.

[5] Lu, S. C-Y., "Collective Rationality of Group Decisions in Collaborative Engineering", *International Journal of Collaborative Engineering*, Vol.1, No. 1/2, pp. 38-74, 2009.

[6] Lu, S. C-Y., Liu, A., "A Synthesis Decision Framework for Early-stage Innovative Design", *21st CIRP Design Conference*, Daejeon, March 27-29, 2011.

[7] Nordlund, M., Tate, D., Suh, N.P., "Growth of Axiomatic Design through Industrial Practice", *3rd CIRP Workshop on Design and the Implementation of Intelligent Manufacturing Systems*, Tokyo, Japan, pp. 77-84, 1996.

[8] Pahl, G., and Beitz, W., *Engineering Design: A Systematic Approach,* 2rd Edition, London: Springer, 1996. ISBN: 3-540-19917-9.

[9] Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. and Lorensen, W., *Object-oriented Modeling and Design,* Prentice-Hall, 1991. ISBN: 0-13-629841-9.

[10] Simon, H. A., *The Sciences of the Artificial*, Cambridge: MIT Press, 1996. ISBN: 0-262-69191-4.

[11] Saaty, T. L., “How to make a decision: The analytic hierarchy process”, *European Journal of Operations Research*, 48, 9–26, 1990.

[12] Suh, N.P., *The Principles of Design*, New York: Oxford University Press, 1990. ISBN: 0-19-504345-6.

[13] Suh, N. P., *Axiomatic Design: Advances and Applications*, New York: Oxford University Press, 2001. ISBN: 0-19-513466-4.

[14] Zeng, Y., "Axiomatic Theory of Design Modeling," *Transaction of SDPS: Journal of Integrated Design and Process Science*, 6(3), pp. 1-28, 2002.