Queensland University of Technology
Brisbane Australia

This is the author's version of a work that was submitted/accepted for publication in the following source:

Staub-French, Sheryl & Nepal, Madhav Prasad (2007) Reasoning about component similarity in building product models from the construction perspective. *Automation in Construction*, *17*(1), pp. 11-21.

This file was downloaded from: http://eprints.qut.edu.au/58418/

© Copyright 2007 Elsevier B.V.

This is the author's version of a work that was accepted for publication in Automation in Construction. Changes resulting from the publishing process, such as peer review, editing, corrections, structural formatting, and other quality control mechanisms may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. A definitive version was subsequently published in Automation in Construction, [17, 1, 2007] http://dx.doi.org/10.1016/j.autcon.2007.02.013

Notice: *Changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published source:*

http://dx.doi.org/10.1016/j.autcon.2007.02.013

Reasoning about Component Similarity in Building Product Models from the Construction Perspective

Sheryl Staub-French [a, *] and Madhav Prasad Nepal [a]

[a] Dept. of Civil Engineering, University of British Columbia, 6250 Applied Science Lane, Vancouver BC, V6T 1Z4, Canada

[*] Corresponding author. Tel.: +1-604-827-5118; Fax: +1-604-822-6901

E-mail: ssf@civil.ubc.ca (Sheryl Staub-French); mpnepal@civil.ubc.ca (Madhav P. Nepal)

Abstract

Identifying the design features that impact construction is essential to developing cost effective and constructable designs. The similarity of building components is a critical design feature that affects method selection, productivity, and ultimately construction cost and schedule performance. However, there is limited understanding of what constitutes similarity in the design of building components and limited computer-based support to identify this feature in a building product model. This paper contributes a feature-based framework for representing and reasoning about component similarity that builds on ontological modeling, model-based reasoning and cluster analysis techniques. It describes the ontology we developed to characterize component similarity that represents the building component, the component attributes, the direction, the range of acceptable variation for geometric attributes, and the degree of variation required to assess component similarity. It also describes the generic reasoning process we formalized to identify component similarity in a standard product model based on practitioners' varied preferences. The generic reasoning process evaluates the geometric, topological, and symbolic similarities between components, creates groupings of similar components, and quantifies the degree of similarity. We implemented this reasoning process in a prototype cost estimating application, which creates and maintains cost estimates based on a building product model. Validation studies of the prototype system provide evidence that the framework is general and enables a more accurate and efficient cost estimating process.

Keywords: Product modeling, Component similarity, Product features, Constructability, 3D CAD, Ontology, Model-based reasoning, Feature recognition, Cluster analysis

INTRODUCTION

Recognizing the design conditions that affect constructability is essential to developing cost-effective designs. While there are many factors that affect constructability, design-specific factors are particularly important because they have the greatest influence on construction cost [1]. Building component similarity is a critical design condition that can have a significant impact on constructability. However, there is little agreement on what constitutes similarity in the design of building components and little understanding of how component similarity might be assessed in a given building product model.

Previous research efforts have recognized the importance of building component similarity (also referred to as uniformity or consistency in design) in constructability reasoning [2], method selection [3,4], productivity modeling [5], and activity sequencing [6]. However, the approaches to date have either represented this concept implicitly in computer code or vaguely in prescriptive statements. In practice, there is limited computer-based support for evaluating the similarity of building components based on practitioners varied preferences and interpretations. Consequently, practitioners today spend significant amounts of time manually interpreting the design to determine whether their particular definition of similarity exists in a given design. There is a need for automated and customizable methods for identifying component similarity in a building product model.

The research presented in this paper addresses this need by providing a formal and flexible way to represent and reason about component similarity. This work is part of an on-going research project that is developing an ontology of features to represent the design conditions that are important for construction. The goal is to develop general ontological models to characterize construction-specific design features in a way that is consistent, unambiguous, and computer-interpretable. We represent component similarity as a product feature of a building product model and use feature recognition to infer its existence in a standard product model.

This paper describes the ontology we developed to represent component similarity, and the generic reasoning process we formalized to evaluate the degree of similarity in a given design. The ontology formalizes a feature-based representation of building component similarity that represents the building component, the component attributes, the direction, the range of acceptable variation for geometric attributes, and the degree of variation required to assess component similarity. The generic reasoning process leverages this project-independent representation to assess the degree of similarity in a given product model based on user-defined criteria. This three step reasoning process employs model-based reasoning and cluster analysis techniques to evaluate the geometric, topological, and symbolic similarities between components, create groupings of similar components, and quantify the degree of similarity. We implemented this reasoning process in a prototype cost estimating application, Activity-based Cost Estimating (ACE), which creates and maintains cost estimates based on a building product model. The system identifies relevant cost-incurring design features, including building component similarity, and adjusts the labor productivity rates and construction methods accordingly to calculate the construction cost.

The following section describes a case study that illustrates different practitioners' criteria for specifying component similarity. Subsequent sections describe the ontology, the reasoning process, and the prototype implementation. Finally, the specific validation studies conducted to date will be discussed.

MOTIVATING CASE

This section describes a case scenario to illustrate the requirements for representing and reasoning about component similarity. The scenario focuses on drywall construction for a building project in Menlo Park, California (Figure 1). One of the authors worked closely with the cost estimators in this project to understand the subtleties for how they assessed component similarity in their cost estimating process. It highlights the different design conditions practitioners consider, and the different techniques they use to characterize the degree of variation in the drywall design.

Figure 1 shows some of the design conditions that were important to the drywall cost estimators in assessing the degree of variation. They focused on these design conditions because they impact construction execution and construction cost. The drywall estimators were concerned with the variety of Wall Types (Figure 1b) since it affected the specific items needed in the cost estimate and the base crew productivity, the variety of Wall Heights (Figure 1c) since it affected the crew productivity and the methods required, and the instances of Wall-Column connections (Figure 1d) since they require additional set up and framing time.

The drywall estimators assessed the degree of variation in different ways depending on the nature of the design condition and the combination of design conditions in the overall design. Consider the estimator's process in determining the productivity rate for metal stud installation and the different types of analysis it required. When considering the degree of variation in the wall heights, the estimator was mostly concerned with the range of wall heights. If the wall heights were within 30 cm (12 in.) of each other, they were similar enough to use the same base productivity rate. When considering the degree of variation in the overall design, the estimator used qualitative measures, such as ‘high’ or ‘moderate’ variation, and adjusted the productivity rate up or down accordingly.

It is often too time-consuming for estimators to manually analyze and interpret all the 2D and related design information to evaluate the similarity of building components in a given design, particularly for large projects. Lacking automated support, practitioners often employ *ad hoc methods* (e.g., estimators may quickly scan the 2D drawings and make rough adjustments to the productivity rate), evaluate component similarity *inconsistently* (e.g., the drywall estimator must remember how he represented and accounted for component similarity to consistently estimate the next project), and rely on *ambiguous measures* (e.g., the estimator looked for ‘high’ or ‘moderate’ degrees of similarity but it is unclear what level of similarity meets this standard).

Practitioners need automated support to formally and thoroughly evaluate the similarity of building components in a given design. The case study highlights the requirements of such a system. To represent and reason about component similarity, it is necessary to:

- Represent numeric (geometric) component attributes (e.g., Wall Height), non-numeric component attributes (e.g., Wall Type), and relationships between components (e.g., Walls ConnectedTo Columns),
- Represent ranges of numeric attributes (e.g., Wall Heights within 30 cm (12 in.)),
- Assess multiple criteria simultaneously (e.g., Wall Type and Height), and
- Consistently and unambiguously evaluate the overall degree of variation (e.g., provide a quantitative measure rather than ambiguous qualitative measures like ‘moderate’ variation).

- Support customization of the criteria to represent practitioners' varied preferences (e.g., the component attributes considered, the range for numeric attributes, and the overall degree of variation),

Although the case study focuses on the application of construction cost estimating, it provides insights into the different project management tasks that can be affected by variations in the design of building components. For example, different wall types require different sizes and types of metal studs, which increase the material handling and decrease the productivity of the metal stud installation. Similarly, walls with a height between 2.43 m and 3.96 m (8 ft. and 13 ft.) require scaffolding for installation whereas walls with a height between 3.96 m and 6.09 m (13 ft. and 20 ft.) require a scissor-lift for installation. These variations in design affect crew productivity and the selection of construction methods, and ultimately impact cost and schedule performance.

BACKGROUND

Many researchers in the AEC industry have recognized the importance of design uniformity and similarity in construction [2-5, 7-9]. Hanna et al. [7] considered uniformity as a critical factor in selecting slab and wall formwork systems. They considered horizontal uniformity to be achievable by satisfying three conditions: regular slab type, identical beam size and location, and regular location and size of cantilevered balconies, whereas vertical uniformity of a building was assumed as one with the same size, height and location of wall from floor to floor. In terms of computer-based implementations, Udaipurwala and Russell [4] developed a rule-based algorithm to infer component uniformity/similarity to aid in the selection of construction methods. However, the approaches to date either represent this knowledge implicitly in computer code (e.g., [2]), vaguely in prescriptive statements (e.g., [3]), or do not reason about a product model (e.g., [4]).

Research on similarity reasoning has been a subject of interest in manufacturing, biology, cognitive science, and information systems for a long time. The most common similarity measures used in manufacturing include group technology, variant process planning, geometric approaches (constructive and boundary models), feature-based approaches, and pattern recognition [10]. The purpose there is to recognize similar machines, classify and index designs, and assess the manufacturability of designs.

We use the manufacturing concept of features to represent and reason about component similarity. We treat component similarity as a *product feature* of a building product model and use *feature recognition* to infer the existence of this feature in a standard product model. Product features are used extensively in manufacturing to describe the geometric forms or entities in a product model that are important in some aspect of the manufacturing process [11,12]. Feature recognition has been extensively researched in the manufacturing industry (e.g., [13,14]). Feature recognition systems can automatically identify features after the part is modeled by using the geometric and topological data from the CAD model. An alternative approach is to use a feature-based design system. This method allows designers to add features as they create the product model, which eliminates the need for feature recognition. However, the feature 'component similarity' is subjective and can be based on a variety of criteria. Consequently, it is unrealistic to expect the designer to add this as a feature to the product model. We developed feature recognition methods to identify component similarity in a standard building product model.

The Industry Foundation Classes (IFCs) developed by the International Alliance for Interoperability are the primary product model exchange standard for the architecture, engineering, construction, and facility maintenance (AEC/FM) industry [15]. The IFCs are a high level, object-oriented data model that support seamless data exchange between different applications. Many CAD vendors for the AEC/FM industry can export IFC-based product models, enabling the sharing of these semantically-rich product models with other software applications. The IFCs define the building element classes and properties, geometry, and the topological relationships between elements. IFC-based product models provide the foundation for interpreting the existence of product features, including component similarity, independent of the CAD application that created the 3D model.

In terms of similarity assessment, similarity measures used in other fields include pattern recognition, methods based on analogy, machine learning, and cluster analysis. Cluster analysis, also known as numerical taxonomy, automatic classification, or typological analysis, is the process of grouping a set of physical or abstract objects into classes of similar objects [16]. Most applications, including construction, require the combination of mixed types of data, including numeric, symbolic, and relational data. Researchers have acknowledged that similarity measures should include nominal and numeric attributes [17] as well as the relationships between objects [18]. We used cluster analysis in our reasoning to create groupings of components based on the similarity of their geometric, symbolic and topological attributes.

Building component similarity assessment can be systematized by developing an ontology that formalizes a feature-based representation of a building product model. Ontologies provide a framework to represent the semantics of data about a certain domain and are used extensively in Artificial Intelligence research [19]. Gruber [20] defines an ontology as an explicit specification of a conceptualization. He refers to a conceptualization as the objects, concepts, and other entities that are assumed to exist in some area of interest and the relationships that hold among them. The two main elements of ontologies are concepts and relations; where concepts are used to define and explain things and relationships order the concepts, often in a hierarchical structure [19]. Ontologies are particularly useful for representing domain-specific knowledge and developing knowledge-based technologies because they provide: (1) a common vocabulary, (2) explication of what has been often left implicit, (3) systematization of knowledge, (4) standardization, and (5) meta-model functionality, where the concepts and relations among them are used as building blocks for the model [21]. We use an ontology to provide a structured and consistent way to represent building component similarity from a construction perspective. We define its vocabulary in terms of objects and attributes such that it enables a knowledge-based program to automatically and systematically identify configurations of building component similarity in a given product model.

FRAMEWORK FOR MODELING COMPONENT SIMILARITY

To understand the subtleties for how practitioners think about the design in the context of assessing component similarity, we reviewed previous research in this area and interviewed construction professionals. We interviewed 14 practitioners from five different construction domains. We interviewed two general contractors and twelve subcontractors that self-perform construction work on drywall, structural concrete, mechanical ductwork, process piping, and electrical systems. We implemented three case studies on two drywall construction projects and one case study on a concrete column construction project. We abstracted the types of design information estimators consider, the different ways estimators quantify the degree of component similarity, and the different steps estimators perform to evaluate component similarity.

We developed a prototype cost estimating application, Activity-based Cost Estimating (ACE), which reasons about component similarity in generating cost estimates from a building product model [22]. ACE identifies relevant cost-incurring design features, including building component similarity, and adjusts the labor productivity rates and construction methods accordingly to calculate the construction cost.

Figure 2 graphically represents our framework for representing and reasoning about component similarity:

(1) **Represent Component Similarity:** Practitioners specify their preferences for defining component similarity in a computer-interpretable template that is based on the ontology we formalized (Figure 2a). Users define component similarity specifications (CSS) according to their preferences. The system represents the instances of this feature generically, independent of a particular project. In the cost estimating application, users also link this feature to specific cost information. This project-independent knowledge is then utilized to compute similarity when the practitioner is ready to create a cost estimate for a project-specific 3D design.

(2) **Identify Component Similarity:** ACE creates a project-specific configuration of component similarity based on the practitioner's generic preferences defined in the CSS. The formal methods we developed reason about the geometric, topological, and symbolic similarities between components in the input 3D model and quantifies the degree of similarity based on the user's preferences (Figure 2b). The result is a project-specific configuration of component similarity customized for the user.

REPRESENTING COMPONENT SIMILARITY

The ontology we developed to represent component similarity allows practitioners to specify their varied preferences for what component properties need to be similar and how much variation is acceptable for component similarity to exist. This work fits into a broader research effort that is trying to characterize the design conditions that affect construction in a feature ontology, which will be discussed next. Subsequent sections will describe the attributes we formalized to describe component similarity, and the computer-interpretable templates we developed to collect this information from practitioners.

Background on Features Ontology for Construction

This section provides some background on the ontology of features we are developing to represent the design conditions that affect construction [23]. This work aims to develop the general language and structure of the model so that it can be populated by a variety of construction experts and be broadly applicable across a variety of construction projects and domains. The ontology classifies the features that affect cost, formalizes attributes to describe each feature type, and represents the sets of features and properties that affect costs for a specific construction domain.

We classified features into the following types: (1) *Component Features* are features that result from components in an IFC-based building product model; (2) *Intersection Features* are features that result from intersections of components; and (3) *Macro Features* are features that result from pre-specified combinations of other features. Component Similarity is a specific class of macro feature. We defined attributes for each feature type to help estimators represent feature instances according to their preferences.

Figure 3 shows the features and attributes currently represented in the ontology. Each component feature represents the estimator's preference for what features and properties influence the cost of its construction using the ‘feature set’ and ‘property set’ attributes. Each intersection feature represents the estimator’s preference for what properties of the feature affect a component’s construction cost using the ‘property set’ attribute and what component intersections are important using the ‘related component’ and ‘relating components’ attributes. Each macro feature represents the estimator’s preference for defining component similarity in terms of the component properties that need to be similar and the amount of variation that is allowed to exist. The attributes of the three feature types enable estimators to represent their varied preferences for naming features, specifying the component intersections that are important to them, defining component similarity, and specifying the features and properties that affect a specific component’s construction costs.

The feature ontology provides the blue-print for the additions and changes needed to transform an IFC-based product model into a product model that is useful to cost estimators of building construction. The next section elaborates on the attributes defined to represent component similarity.

Attributes Formalized to Represent Component Similarity

This section describes the attributes currently formalized in the ontology to characterize the feature *component similarity*. The attributes provide a formal way to specify the different types of component properties (e.g., geometric, symbolic and relational attributes) to be evaluated and the direction of analysis (e.g., across a single floor or multiple floors). The ontology also provides a way to characterize the degree of similarity at the component level (e.g., Wall Height ± 30 cm (12 in.)) and at the system level (e.g., 10% variation).

Attribute 1: Component Class:

The component class being evaluated for similarity, such as walls and columns. The components defined in an IFC-based product model are currently represented.

Attribute 2: Component Properties

The component properties (or property) of the component class (Attribute 1) that will be compared to determine whether the components are similar. The case examples demonstrated that when assessing component similarity, practitioners consider geometric, topological, and symbolic component properties. Consequently, we classify component properties into the following types:

- Geometric component attributes: Numeric attributes that are based on the geometry of the component (e.g., Height, Width, and Length).
- Symbolic component attributes: Non-numeric attributes that characterize symbolic aspects of a component (e.g., Type, Color, Fire-Rating).
- Relational component attributes: Attributes that specify explicit relationships between components (e.g., Walls Connected-to Columns).

The classification of component attributes facilitates the evaluation of component similarity. Table 1 shows the different component attributes currently implemented in the ontology based on the different case studies conducted to date. The component attributes listed are either represented explicitly in an IFC-based product model or can be derived from an IFC-based product model.

Attribute 3: Geometric Property Variation

The acceptable variation in the value of the geometric component properties. For example, if a practitioner specifies 5 cm (2 in.) for the property variation of the property ‘height’, then the practitioner views wall #1 as similar to wall #2 if its height is at most 5 cm (2 in.) shorter or taller than wall #2. This attribute accounts for the fact that when practitioners evaluate geometric properties they require some allowance for deviation. For example, the column heights on a floor might vary by a centimeter or two because of a sloping slab or perhaps because of an error in the design drawings. However, from the practitioners perspective a few centimeters does not mean that they are not similar enough to meet their standards. If these minor variances in geometric properties are not explicitly considered, the computer analysis might falsely classify components as dissimilar.

Attribute 4: Direction

The ontology represents the direction for which component similarity will be assessed as either 'horizontal' or 'vertical.' The horizontal direction represents similarity across a single floor and the vertical direction represents similarity across floors. To represent component similarity in both directions, practitioners would need to apply constraints for each direction when setting up the analysis.

Attribute 5: Component Variation

The overall variation of the components allowed to achieve component similarity as a function of a maximum and minimum percentage. In the drywall case example, the practitioner specified a minimum of 75% and a maximum of 100% to represent an 'ideal' degree of similarity for the optimal productivity. When evaluating the overall variation of all the building components design, practitioners also require some allowance for specifying degrees of similarity. The intention here is to establish a way of measuring the degree of similarity such that it is quantifiable, explicit, and consistent. Consider Hanna and Sanvido's [3] measure of 'moderate variation' in their guidelines for selecting formwork systems. Using such terminology is vague and ambiguous and does not facilitate a quantifiable analysis. We tried to provide a flexible way to specify such ranges of variation. We currently use percentages to specify the degree of variation because it allows the user to specify the range independent of a particular unit of measure.

Computer-interpretable Template for Specifying Component Similarity

We developed templates in the ACE prototype that implements the attributes currently represented in the ontology to capture different practitioners' preferences for defining component similarity (Figure 4). Practitioners create instances of Component Similarity Specifications (CSS) using the template. The implementation is interactive, allowing practitioners to specify component similarity according to their preferences.

Practitioners specify the properties of the component that need to be evaluated for similarity using the 'Similar Component Properties' attribute and the degree of similarity that needs to exist using the 'Component Variation' and 'Property Variation' attributes. Practitioners can use these attributes to represent a variety of definitions for component similarity. For cost estimating, they can link a CSS to different cost estimating information. Specifically, users can link the CSS to a specific crew's productivity rate and adjust the productivity rate for a specific degree of similarity (Figure 4), or they can link it to a specific construction method to constrain its availability. This knowledge is represented generically, independent of a particular project, and can be reused from project to project to compute similarity in any given 3D model. The CSS instances defined by the user drives the similarity evaluation process.

REASONING ABOUT COMPONENT SIMILARITY

As illustrated in Figure 2b, our reasoning process involves the following three steps: (1) Identify Relevant Components and Attributes, (2) Compute Similarity between Components, and (3) Group Similar Components and Quantify Degree of Similarity. We will explain each of these three steps in subsequent sections using the drywall case (Figure 1) and the sample product model data shown in Table 2. The drywall estimator is primarily interested in understanding the degree of variation of Wall Types, Heights, and Column Connections (Figure 1b-d). We will discuss the similarity computation as if these constraints are considered independently (e.g., Wall Type is the only property specified in the CSS) and simultaneously (e.g., Wall Type, Height, and Column Connections are all properties specified in the CSS). Figure 4 shows the relevant CSS for the Wall Height constraint.

Identify Relevant Components and Attributes

The input to the reasoning process is a building product model that explicitly represents building components, attributes of components, and relationships between components. In our prototype implementation, we extract the product model data from a live 3D CAD model that represents this information similar to the IFC standard [23]. Therefore, the reasoning process uses model-based reasoning to identify the relevant geometric, topological, and symbolic data from the input product model. In this step, the relevant components are extracted (e.g., all the Interior Walls for the single floor) and the relevant attributes are identified and computed.

In many instances, the relevant component information is explicitly represented in a standard product model (e.g., the wall attributes for ‘Length’ and ‘Height’), which simplifies the reasoning process significantly. However, in certain cases the relevant component attributes are either not explicitly defined by the IFCs or may not be explicitly represented in the 3D model and corresponding IFC output. For example, the IF's do not define 'curvature' as an attribute of the wall. The curvature can be deduced based on the geometry of the wall or it can be represented in an extended property set. Similar issues arise when dealing with component connections. Although the IFCs provide a way to explicitly represent component connections, the specific connections represented in the IFC output of a 3D model depend on the drawing methods employed. For example, consider the connections between the walls and columns illustrated in Figure 1d. Since the structural and architectural models are typically developed by different disciplines and companies, the physical relationships between objects in these different models are typically not explicit. To identify these types of implicit relationships requires the application of conflict detection methods (e.g., to identify intersecting components like the walls and columns) or geometric reasoning methods (e.g., to identify adjacencies). We have tried to avoid these issues by explicitly representing these attributes and relationships in the 3D model whenever possible.

Evaluate Similarity between Components

The CSS (Figure 4) provided by the user defines the parameters that drive the similarity evaluation process. Specifically, the CSS dictates the specific component properties considered, the range of acceptable geometric variation, and the acceptable degree of component variation.

The components (objects) in a building product model can be distinguished by their various attributes, which can be both quantitative and qualitative in nature. Our data set contains relational data (binary variables), geometric data (interval-scaled variables), and symbolic data (nominal variables). As our data set contains mixed variables, we referred to the approaches described by Kaufman and Rousseeuw [16] to compute similarity between objects. To work with mixed variables, one can perform separate cluster analyses, treat the different variables as a single type, or combine the different variables into a single proximity matrix [16]. We treat the different variables as nominal or binary variables and use the simple matching approach (Equation 1), which is the most common way to measure the similarity between two objects characterized by these types of variables. For geometric attributes, we consider a range rather than a specific number based on the user preference defined by the ‘Geometric Property Variation’ attribute in the CSS. Therefore, to identify matches among geometric attributes, we compare the available ranges for the two objects and if the attribute values fall within the acceptable range, the objects are considered similar for that attribute, yielding a True or False result.

The simple matching approach expresses similarity as a coefficient, which looks for the percentage of matches between objects *i* and *j* [16]:

$$s\,(i, j) = \frac{u}{p} \qquad (1)$$

where *u* is the number of matches, that is, the number of variables for which objects *i* and *j* happen to be in the same state and *p* is the total number of variables.

Similarity coefficients indicate how similar two objects *i* and *j* are, where the more objects *i* and *j* are alike, the larger *s (i, j)* becomes [16]. Similarity *s (i, j)* typically takes on values between 0 and 1 to indicate various degrees of resemblance, where 0 means that *i* and *j* are not similar at all and 1 reflects maximal similarity. For all objects *i* and *j*, it is assumed that the following conditions hold:

$$\textbf{(S1)}\ s\ (i, j) = 0 \leq s\ (i, j) \leq 1 \tag{2}$$

$$\textbf{(S2)}\ s\ (i, j) = s\ (i, j) = 1 \tag{3}$$

$$\textbf{(S3)}\ s\ (i, j) = s\ (j, i) \tag{4}$$

Since we were trying to identify the components that meet the criteria specified in the CSS, we primarily dealt with Equation (2) where similarity along a specific attribute was either True or False. We did not consider the relative similarity of attribute values (e.g., the similarity of Wall Types P-2 and P-2a), we weighted all component attributes equally (e.g., Wall Height and Type were considered of equal importance), and we assumed there is no correlation between attributes (e.g., the correlation between a Wall's Height and Width). This was the basis for computing similarity between pairs of objects.

6.1 Group Similar Components and Quantify the Degree of Similarity

We grouped similar objects based on the similarity of each of the variables, as illustrated in Figure 5. We assume all objects are similar at the start because at this point, we just have a grouping of components of the same type (e.g., a grouping of Walls) and we have not considered the component attributes yet. Then we cycle through each of the attributes based on the priorities set by the practitioner to create a single grouping of similar components. We then compute the overall degree of similarity by summing up the number of objects in the similar grouping and dividing by the total number of objects.

As evident in the case data, Walls 1, 2, 4, and 5 are similar if focusing on the single attribute Wall Type, yielding an overall similarity of 80%. This would be the computed degree of similarity if Wall Type was the only property specified in the CSS. If the attribute Wall Height is added as an additional constraint in the CSS, then Walls 1, 2, and 5 are grouped yielding an overall similarity of 60%. Finally, Walls 2 and 5 are 100% similar based on the three attributes

Wall Type, Wall Height, and Column Connection, resulting in an overall similarity of 40%. The judgment of Table 2 also intuitively supports the results of this computation.

Based on the practitioner's criteria for similarity expressed in the CSS, we see that 80% of the walls satisfy the Wall Type constraint and meet the overall component variation criteria of 75-100%. However, when the additional constraint of Wall Height ±15 cm (6 in.) is added, only 60% of the walls satisfy this additional constraint and therefore do not meet the practitioner's criteria for acceptable component variation. Naturally, when the third constraint of Column Connections is added, only 40% of the walls are similar, which falls far short of meeting the practitioner's criteria.

Although this is a simple example, it illustrates the flexibility and ease of use of our approach. Practitioners can consider single or multiple attributes to easily create a variety of component similarity definitions. It also demonstrates the importance of choosing the relevant component properties carefully. If too many properties or inappropriate combinations of properties are selected by the practitioner in the CSS, then the system will detect limited or no similarity which will likely be of little use. In our current implementation, we do not help the user to pick the appropriate properties. Practitioners can also easily specify different ranges of acceptable variation, which over time, will help them to better understand how component similarity impacts their construction environment.

The output of this process is used in our prototype cost estimating application ACE. If the computed degree of similarity meets the specification of the user in the CSS, ACE adjusts the related labor productivity rates or assigns the relevant construction methods accordingly when calculating the construction cost [23].

VALIDATION

We used ACE to validate our framework for representing and reasoning about component similarity. We performed four validation tests, including a Charrette test and three retrospective tests [24]: (1) Charrette test with eight industry practitioners estimating interior wall construction costs, (2) Retrospective test case of estimating interior wall construction costs on the Sequus Pharmaceuticals project [25], (3) Retrospective test case of estimating interior wall construction costs on a DPR Office project, (4) Retrospective test case of estimating concrete column construction costs on the Bay Street Emeryville Project. These tests provide evidence for the power and generality of our framework, as described below.

To demonstrate generality, we wanted to show generality across component types and user types. We modeled costs for two different component types (interior walls and concrete columns) in three retrospective test cases. Different configurations of component similarity are required by practitioners estimating costs for these domains, including different component properties and different degrees of variation. We also demonstrated that 13 different estimators could specify their preferences for defining component similarity for the different test cases.

To demonstrate power, we wanted to show that our approach enabled cost estimators to generate and maintain cost estimates more accurately, consistently, and efficiently. To assess the accuracy of the estimates, we evaluated the *level of completeness* of estimates generated by 13 estimators using ACE and compared them to estimates generated by the same estimators using Timberline's state-of-the-art Precision Estimating (PE) software [26], an industry standard for cost estimating software. We used level of completeness to measure the extent to which estimators accounted for the cost impacts of features explicitly (including component similarity). If estimators used ad hoc methods or overlooked the cost impact of features, they received a lower score for completeness. The results of the validation tests demonstrate that practitioners could generate and maintain more complete cost estimates in ACE than the state-of-the-art process. Estimators could generate and maintain cost estimates that are less ad hoc and contain fewer omissions than estimators using state-of-the-art tools. The Charrette test demonstrates that practitioners using ACE were able to more consistently identify the correct cost impact and identify the cost impacts 17% faster using ACE when compared with the state-of-the-art process. Therefore, the four validation tests demonstrate that practitioners could account for the cost impact of features, including component similarity, more completely, consistently, and quickly using ACE than the same practitioners using state-of-the-art tools.

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE RESEARCH

This paper contributes a feature-based framework for representing and reasoning about component similarity that builds on ontological modeling, model-based reasoning and cluster analysis techniques. We formalized an ontology that represents the building component, the component attributes, the direction, the range of acceptable variation for geometric attributes, and the degree of variation required to assess component similarity. A computer implementation of the ontology enables practitioners to represent their varied preferences for defining component similarity generically and consistently. We developed a generic reasoning process that leverages this project-independent representation to identify project-specific instances of component similarity in a given 3D product model. This three step reasoning process evaluates the geometric, topological, and symbolic similarities between components, creates groupings of similar components, and quantifies the degree of similarity. We provide evidence that this framework is general and enables a more complete, consistent, and efficient cost estimating process.

The framework presented in this paper is limited in many ways. Additional work is needed to account for the subtle similarities between components that are not accounted for by considering the attribute values. For example, Wall Types ‘P-2’ and ‘P-2a’ are very similar but if you only look at the attribute value, this relative similarity does not get addressed. Practitioners should also have the ability to weight certain component attributes more heavily than others when considering multiple attributes. Additionally, our current approach of relying solely on the user to identify the relevant properties and degrees of variation could be improved by allowing some combination of user-driven and system-driven identification of the relevant information for a given product model. Finally, this work should be extended to provide feedback to designers so that they can better understand how practitioners view component similarity and to optimize the degree of similarity in their designs.

Automating the detection of construction-specific design features, like component similarity, has the potential to significantly improve the efficiency of the project delivery process. Estimators could provide cost feedback in significantly less time. Project teams could perform what-if analyses on different designs and explore a larger variety of design alternatives to identify the lowest cost design. Practitioners could provide feedback to designers on the specific features that impact construction costs. Hence, project teams can leverage feature-based product models to develop more cost-effective and constructable designs in less time.

REFERENCES

[1] B.C. Paulson, Jr., Designing to reduce construction costs, Journal of the Construction Division, 102 (4) (1976) 587-592.

[2] M. Fisher, Constructability input to preliminary design of reinforced concrete structures, Technical Report, Center for Integrated Facility Engineering, Stanford University, CA, 1991.

[3] A.S. Hanna and V.E. Sanvido, Interactive vertical formwork selection, Concrete International: Design and Construction, 12 (4) (1990) 26-32.

[4] A.Udaipurwala and A.D. Russell, Hierarchical clustering for interpretation of spatial configuration, in: Proc. of the 2005 ASCE Construction Research Congress, San Diego, CA, 2005, 556-571.

[5] H.R. Thomas and I. Zavrski, Construction baseline productivity: theory and practice, Journal of Construction Engineering and Management, 125 (5) (1999) 295-303.

[6] D. Echeverry, W.C. Ibbs, S. Kim, Sequencing knowledge for construction scheduling, Journal of Construction Engineering and Management, 117 (1) (1991) 118-130.

[7] A.S. Hanna, J.H. Willenbrock, V.E. Sanvido, Knowledge acquisition and development for formwork selection system, Journal of Construction Engineering and Management, 118 (1) (1992) 179-198.

[8] M. Skibniewski, T. Arciszewski, K. Lueprasert, Constructability analysis: a machine learning approach, Journal of Computing in Civil Engineering, 2 (1) (1997) 8-17.

[9] O.O. Ugwu, C.J. Anumba, A. Thorpe, The development of cognitive models for constructability assessment in steel frame structures, Advances in Engineering Software, 35 (2004) 191-203.

[10] A. Elinson, D.S. Nau, W.C. Regli, Solid modeling and applications, in: Proc. ACM Symposium, Atlanta, GA, 1997, 297-310.

[11] J.J. Cunningham and J.R. Dixon, Designing with features: the origin of features, ASME Computers in Engineering Conference, San Francisco, CA, 1988, pp. 237-243.

[12] J.J. Shah, Assessment of features technology, Computer-Aided Design, 23 (5) (1991) 331-343.

[13] M.R. Henderson, Extraction of feature information from three dimensional CAD data, PhD Thesis, Purdue University, 1984.

[14] R. Gadh and F. B. Prinz, Recognition of geometric forms using the differential depth filter, Computer Aided Design, 24 (1992) 583-598.

[15] International Alliance of Interoperability (IAI), IFC 2x Extension Modeling Guide, Available from http://www.iai-na.org/, 2001. <accessed on August, 22, 2005>

[16] L. Kaufman and P.J. Rousseeuw, Finding Groups in Data: an Introduction to Cluster Analysis, John Wiley & Sons, Inc., 1990.

[17] C. Li and G. Biswas, Conceptual clustering with numeric and nominal mixed data - a new similarity based system, IEEE Transactions on KCE, 1998.

[18] G. Bisson, Why and how to define a similarity measure for object-based representation systems, Towards Very Large Knowledge Bases, Amsterdam, IOS Press, (1995) 236-246.

[19] H. Scholten and A. Beulens, Ontologies to structure models and modeling tasks, 16th JISR-IIASA Workshop on Methodologies and Tools for Complex System Modeling and Integrated Policy Assessment, Laxenburg, Austria, July 15–17, 2002.

[20] T.R. Gruber, Toward principles for the design of ontologies used for knowledge sharing, International Journal of Human Computer Studies, 43 (5–6) (1995) 907–928.

[21] R. Mizoguchi, Ontological engineering: foundation of the next generation knowledge processing, in Proc. 1st Int. Conf., Web Intelligence and Intelligent Agent Technologies, Maebashi City, Japan, (2001) 44–57.

[22] S. Staub-French, M. Fischer, J. Kunz, B. Paulson, Jr., A generic feature-driven activity-based cost estimation process, Advanced Engineering Informatics, 17 (1) (2003) 23-39.

[23] S. Staub-French, M. Fischer, J. Kunz, B. Paulson, Jr., K. Ishii, A feature ontology to support construction cost estimating, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, 17 (2), (2003) 103-114.

[24] S. Staub-French, Feature-driven activity-based cost estimating, PhD Thesis, Stanford University, Stanford, CA, 2002.

[25] S. Staub-French, and M. Fischer, Industrial case study of electronic design, cost, and schedule integration, Technical Report, vol. 122, CIFE, Stanford University, 2001.

[26] Timberline Software Company, Precision Estimating Extended and CAD Integrator, Users Documentation, Beaverton, Oregon, 2001.

Figure Captions

1. Fig. 1: 3D Model of drywall case study (a), and design conditions that impact drywall construction: Variety of Wall Types (b), Variety of Wall Heights (c), and Instances of Wall Connections (d).
2. Fig. 2: Framework for (a) representing and (b) reasoning about component similarity to create project-specific configurations of component similarity.
3. Fig. 3: Feature ontology that represents the attributes of the three feature types and features currently implemented for wall and column components.
4. Fig. 4. Template for specifying component similarity and example practitioner's preference that 75-100% of the walls have wall heights ±15 cm (6 in) for component similarity to exist.
5. Fig. 5. Groupings of similar components based on the similarity of the different attributes and the degree of similarity.

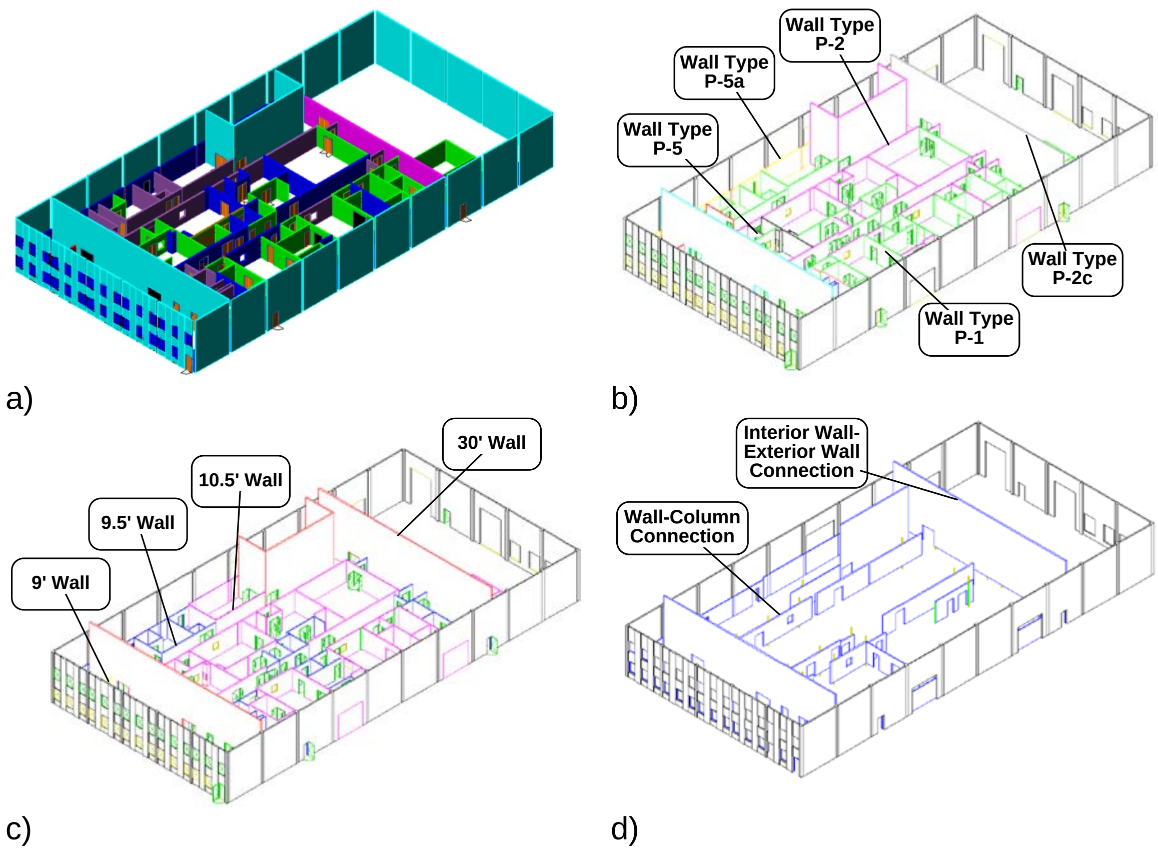

Fig. 1: 3D Model of drywall case study (a), and design conditions that impact drywall construction: Variety of Wall Types (b), Variety of Wall Heights (c), and Instances of Wall Connections (d).

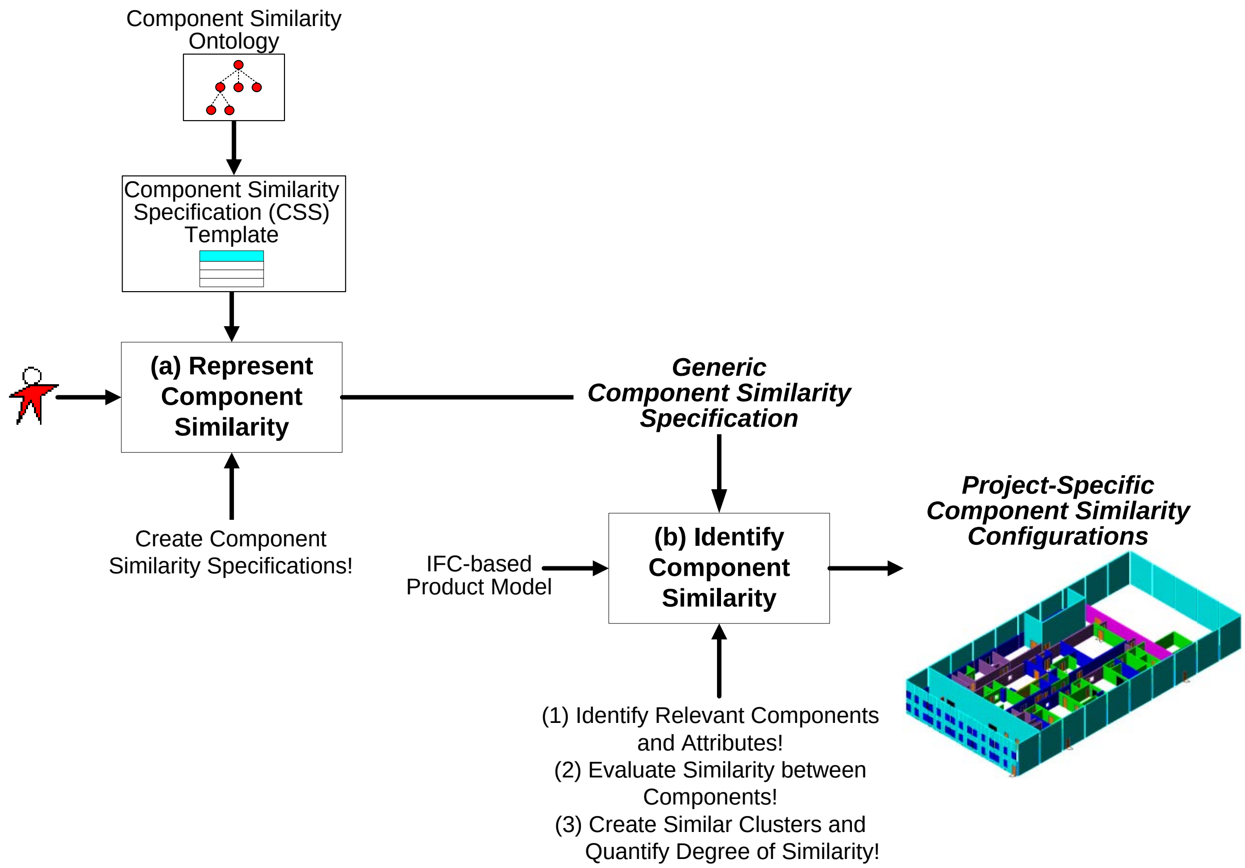

Fig. 2: Framework for (a) representing and (b) reasoning about component similarity to create project-specific configurations of component similarity.

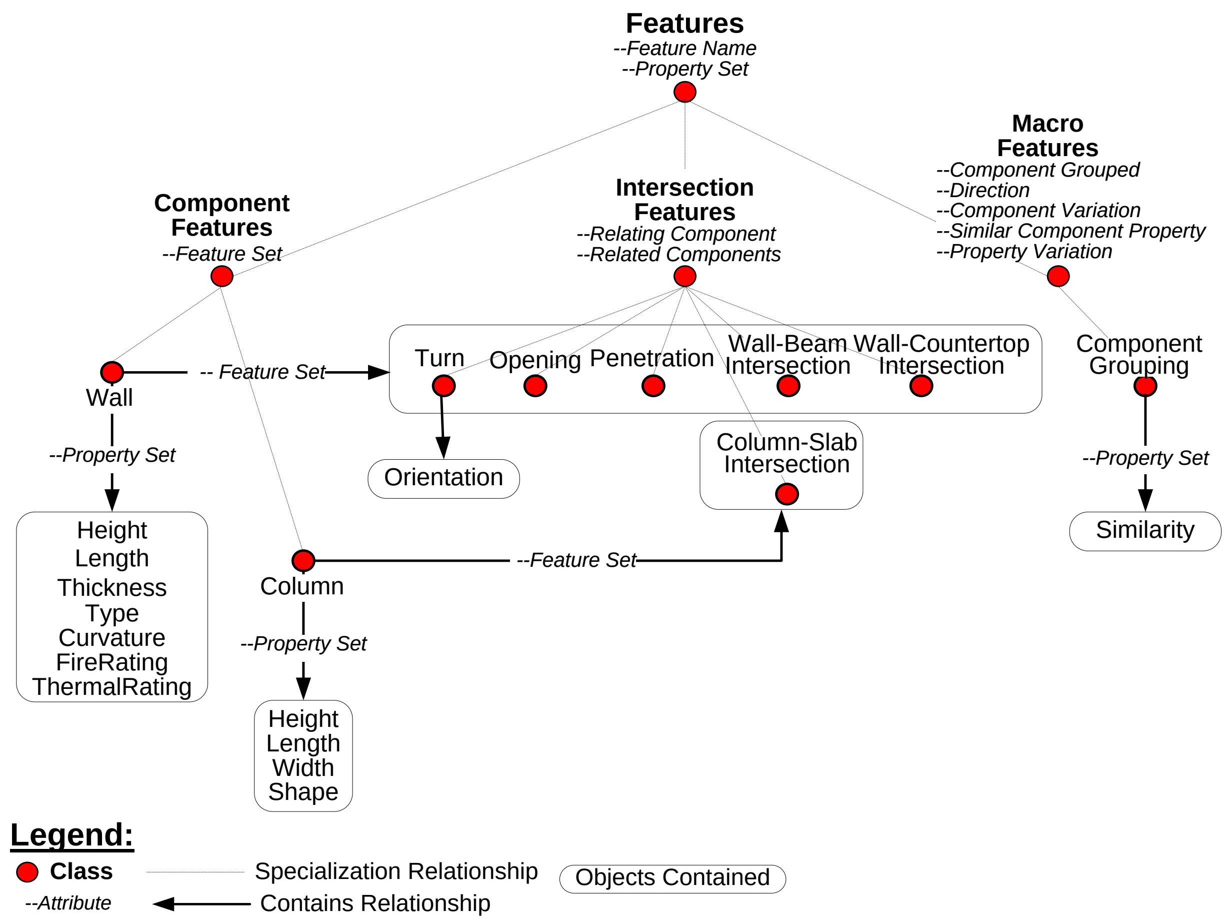

Fig. 3. Feature ontology that represents the attributes of the three feature types and features currently implemented for wall and column components.

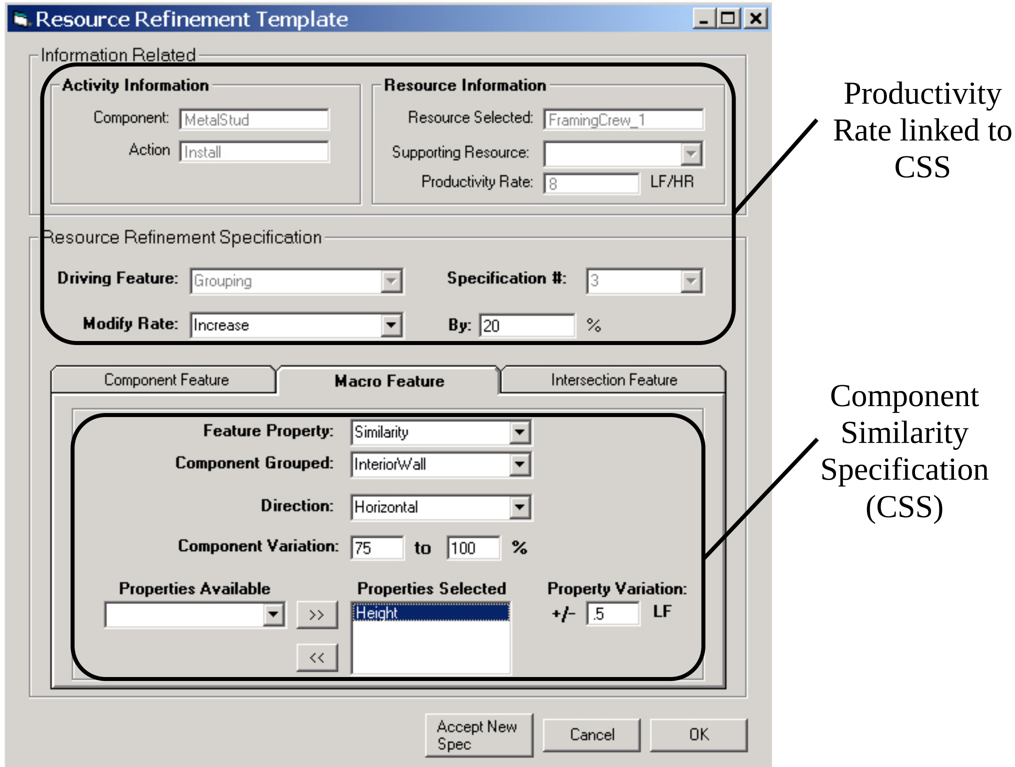

Fig. 4. Template for specifying component similarity and example practitioner's preference that 75-100% of the walls have wall heights ±15 cm (6 in.) for component similarity to exist

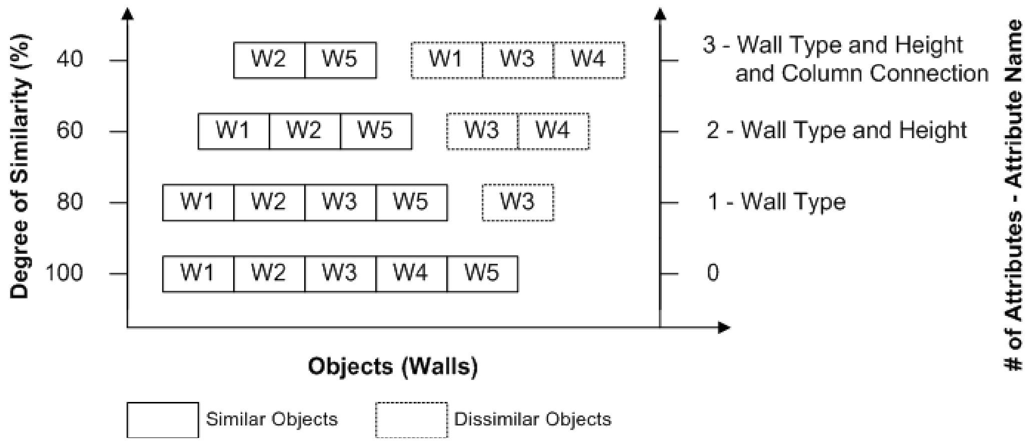

Fig. 5. Groupings of similar components based on the similarity of the different attributes and the degree of similarity

Table 1. The component attributes currently implemented in the ontology

Component class	Component Attributes		
	Geometric	Symbolic	Relational
(1)	(2)	(3)	(4)
Interior Walls	Length Thickness Height	Type Curvature FireRating AccousticRating ThermalRating ExternalWall	ConnectedTo DecomposesInto HasOpenings
Concrete Columns	Length Width Height	Type Shape	ConnectedTo DecomposesInto

Table 2. A Sample dataset of drywall object instances from the motivating case

Objects (1)	Wall type (2)	Frame height (m) (3)	Connectivity to column (4)
1	P-1	3.04	Yes
2	P-1	2.89	No
3	P-2	9.14	Yes
4	P-1	3.65	No
5	P-1	3.04	No