A Knowledge-Based Framework for Quantity Takeoff and Cost Estimation in the AEC Industry Using BIM

S. Aram[a], C. Eastman[a] and R. Sacks[b]

[a]College of Architecture, Georgia Institute of Technology, USA
[b] Civil and Environmental Engineering, Technion - Israel Institute of Technology, Israel
E-mail: shiva_aram@gatech.edu, charles.eastman@coa.gatech.edu, cvsacks@techunix.technion.ac.il

Abstract

An important set of information provided through Building Information Modeling (BIM) platforms are quantitative properties of design elements and assemblies. The capability to extract or deduce such quantitative properties from explicit and implicit model information is essential for bidding, procurement, production planning, and cost control activities in the AEC projects. Current solutions for quantity take off (QTO) and cost estimation (CE) are developed based on the assumptions that the design models are suitable, contain adequate information to perform these tasks efficiently and accurately. In practice often these criteria do not exist in the models that cost estimators receive. Many estimators, engineers and managers distrust BIM operations as a result or find it difficult to adopt a BIM-based preconstruction process. This leads to a cumbersome, manual and error-prone QT and CE process currently used by most construction companies. In order to overcome these shortcomings, we have developed a framework for a knowledge-based system to perform model based QTO and CE. This framework includes domain, reasoning, task and interface layers. This paper reports on the progress on an ongoing research effort which so far mostly focused on developing a domain layer and rule libraries for the reasoning layer. The domain layer contains a knowledge base which along with rule libraries were developed by acquiring and representing domain experts' knowledge. The rule libraries include modules of rules to infer knowledge about different product features. The inferred knowledge will enable providing and representing model information in a compatible format for QTO and CE tasks. It facilitates filtering, grouping and representing feature information provided in design models based on criteria that determines their true cost behavior. Finally, this knowledge will enable forecasting the properties of product features absent from design models. Examples are drawn from various fields inside and outside of the AEC industry, with a focus on the precast concrete industry.

Keywords
Knowledge based systems, knowledge inference, quantity take off, cost estimation, precast concrete

1 Introduction

Efficient and accurate quantity take off (QTO) and cost estimation (CE) are pivotal to a project's success. They are knowledge-intensive [1]; they are the prerequisites to many other activities in a project from budgeting, bidding and contracting to value based design, production planning and budget control; they require extracting information based on the knowledge of domain experts about the rules and processes throughout the products and projects lifecycle. There are commercial software products available that attempt to semi-automate these tasks through augmenting the quantitative information elicited from design models, creating pre-structured yet customizable cost databases and reducing repetitive aspects of these tasks [2].

Based on our study, QTO software products need to maintain three conditions for their successful performance (i) architectural and structural design models to be readily *suitable* for quantity takeoff and cost estimation; (ii) all the needed information to be *quantitative* in nature; (iii) designers' models to contain *complete* information needed for these tasks. In practice these conditions are rarely met. The focus here is not on users' modeling practices and their use of correct modeling methods. Yet even when designs are correctly modeled:

1. Categories of contained information in models developed by designers and constructors and the way the information items are modeled and represented are different, as these models serve different purposes. Two examples are Cast-In-Place (CIP) and precast reinforced concrete products where the units of quantity take off and cost estimation are each concrete placement breaks and a product piece, respectively. However, these units often are not distinguished in models. This difference leads to rework and often for QTO and CE purposes, different construction parties have to create their own models from scratch.
2. The main focus of these solutions are eliciting and enhancing a set of standard quantities like volume, surface area, etc. for different products. The problem is that (a) each product type needs elicitation of a specific set of design properties for QTO and CE which can only be determined based on that product's supply chain, (b) sometimes the properties that impact cost of a product are not inherently quantitative. Current systems either don't elicit information about these properties from design models or they are represented as raw data and can't provide the user with the insight needed for decision-making. An example is product shape. Different shape parameters that impact the cost and in what value ranges their cost relationships and behavior change should be identified.
3. The detailed design with complete information for rigorous cost estimation are developed late in the project lifecycle and usually for fabrication and production of products. For instance due to high time and cost required, many features of reinforced concrete products like connections that are important for accurate cost estimation of reinforced concrete products are often designed and modeled after the companies are contractually bound to the project. Currently QTO and CE experts mostly rely on their judgment and rules of thumb which are developed based on historical information and for unusual situations, they seek expertise of structural designers, plant managers, erectors and others on a case by case basis. This process is manual, time consuming and error prone.

These issues create considerable technical drawbacks for efficient and accurate model-based quantity takeoff and cost estimation. Our studies have shown that currently the QTO and CE processes employed by most construction subcontractors, where a detailed QTO and CE is required, is generally manual and a majority of companies only use 2D drawings rather than 3D parametric models.

In an attempt to overcome these limitations, we outlined development of a framework for a Knowledge-Based System (KBS) to identify, define and retrieve the minimum set of model information required for quantity takeoff and cost estimation of building systems. The example building system that we have selected to implement a proof of concept is precast concrete. However, the developed methodology and structure of this framework have been defined to address broader applications and is adaptable to other building systems.

This framework is designed in a way that it addresses the three above mentioned shortcomings. We study, identify and embed the rules to provide and represent information in BIM platforms in a compatible form for QTO and CE purposes. The specific set of design features and their properties, both qualitative and quantitative, that impact the cost of a project are identified. The criteria to categorize and represent these features in groups are defined, based on parameters and their value ranges where their cost relationships change. Knowledge of domain experts is elicited, and will be codified and embedded to forecast the properties of design features required for QTO and CE tasks but absent from design models (e.g. connections) with acceptable accuracy. The complete method will provide estimators with the design-related information required to perform a model-based cost estimation in an efficient and semi-automated way.

2 Research on Knowledge-Based Systems for QTO and CE

Knowledge-Based Systems (KBS) have emerged from the Artificial Intelligence (AI) field and are employed for numerous purposes in various industries. KBS are systems that acquire, represent and process data, information and knowledge to generate knowledge. Unlike traditional information systems they can act as decision makers and serve like an expert on demand [3, 4].

Several research efforts [5, 6, 7, 8] have developed knowledge-based systems for product and project cost estimation purposes. Some of these systems were developed both as a decision-making support system for choosing the manufacturing process, machines and material of products and as a cost estimation solution based on the selected options. For example, Chan & Lewis [6] developed a knowledge-based system incorporating product design, process and cost knowledge into inference engines used for material and process selection and ultimately for cost estimation.

An example in the manufacturing industry is the system developed by Shehab & Abdalla [5] for modeling cost of machining components as well as molded components. The system's inputs include a material, a mold and a processing database as well as geometric and feature data of the product design model. Domain knowledge was represented in an expert system toolkit through frames and rules like material selection rules and manufacturing process and tool selection rules based on various characteristics such as material cost, product functionality and machine availability. Based on the system's recommended process, the product's manufacturing cost was estimated. While some product features like number of cavities and surface finish were factored in the estimated cost, it is not clear how qualitative aspects like shape complexity were contributed to the cost model.

A diverse team sponsored by the National Institute of Standards and Technology sponsored Advanced Technology Program (NIST ATP) developed the Federated Intelligent Product EnviRonment (FIPER) [9] knowledge-driven environment for concurrent engineering to reduce cost of product development. In FIPER product cost information is integrated with the knowledge base. Koonce et al. [10] developed a cost with the goal of providing an integrated web-based estimation tool in which they used the design data provided by FIPER at different stages of design completion. They integrated the design data with a cost engine consisted of Work Breakdown Structure (WBS) elements and element attributes that determine the cost of an element using a hierarchical structure for attribute inheritance.

Knowledge-based systems have been developed for various purposes for the Architecture, Engineering and Construction (AEC) industry as well. For the cost estimation domain, Staub-French et al. [1] proposed a reasoning process based on cost estimators' knowledge to represent and apply their rationale about impact of design features on cost estimation. This process customizes the activities and allocation of resources to each activity to account for project-specific features. Lee et al. [11] developed a framework that uses an ontology designed for work conditions and work items in tiling and through reasoning rules automatically selects the most appropriate work item. The inference process is designed based on knowledge of an expert and the selected work items are then used for cost estimation. In both of these efforts the focus has been on developing an ontology to represent different design and construction conditions that affect the cost of a project.

The reviewed KBSs all assume that product models used for cost estimation include all the information about feature properties that impact projects' cost and that the unit of products represented in product design models fit the cost units of manufacturers. In other words, they only extract information represented explicitly in design models, but cannot modify the design to reflect the fabrication and installation units critical for cost estimation. They do not anticipate product features missing from design until very late stages of a project nor attempt to enhance the information retrieved from design models to contribute to a project' cost estimation.

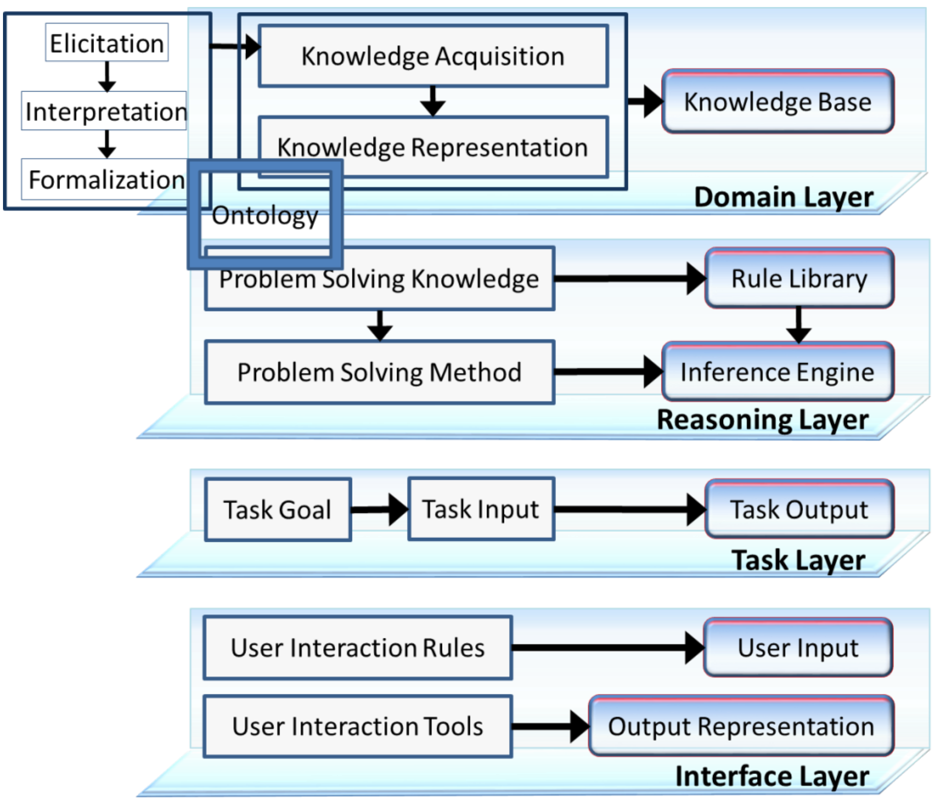

Figure 1. Knowledge based systems structure

These systems would only work under ideal situations when late project information is available early in the project for design entities and is represented in design models, which is relatively rarely the case.

Many of the principles used in the previously developed KBSs can be used in the AEC industry and we use them in our framework. The proposed KBS in this paper aims to build on those frameworks and modify and improve them to depict real work environments. This is achieved by designing a framework to adjust design models and make them suitable for cost estimation without the need for re-doing the model. The key extension is to infer the knowledge critical for accurate cost estimation about missing design features. Thereby the proposed system attempts to enhance the knowledge extracted from design models and to automate the current mostly manual and time-consuming QTO and CE process.

3 Knowledge Based Systems Architecture

The two major components of a KBS architecture include a knowledge base and a reasoning engine [3]. Some researchers have also included a task [12, 13] and a user interface layer [14, 15] as essential and separate components of a KBS structure. Figure 1 illustrates structure of a knowledge-based system.

Domain Layer: Domain layer consists of a knowledge base which is a repository that represents the knowledge acquired from various domains and represented using different representation tools. Knowledge acquisition and representation deal with content and format of knowledge respectively and enhance availability and usability of knowledge [14]. Various textual, graphical and computer-interpretable knowledge representation conventions and tools have been developed to standardize knowledge modeling in different domains. Examples include UML and family of IDEF languages [16].

A knowledge base represents the acquired domain knowledge using an ontology. Ontologies, originally defined by Gruber [17] as “explicit specification of a conceptualization”, are fundamental for sharing and reusing knowledge. An ontology specifies a vocabulary - set of representable objects, their properties and relationships – for a universe of discourse. KBSs model their domain of interest through explicit abstraction hierarchies and rules about their relations that comprise an ontology. Shared ontologies tie modules of a KBS and are essential for communication and reuse of knowledge among different modules of one knowledge base and for integrating knowledge base of separate KBSs [13].

Reasoning Layer: The reasoning layer includes modules of rule libraries and inference engines. Reasoning processes in this layer are outlined by utilizing the concept of a Problem-Solving Method (PSM) which specifies the logics behind the reasoning processes. A PSM determines required inference actions, their dependencies and sequence as well as role of each acquired knowledge piece, namely *observables, abstract observables, solution abstractions* and *solutions* to reach a specific goal [12]. Notion of a shared ontology facilitates implementation of a modularized structure for the reasoning layer where different modules computationally work as an integrated whole.

Task Layer: While hierarchy and relations of tasks are defined in the reasoning layer, a finer decomposing of tasks to the goal, required input, expected output and the strategy applied to generate the output is provided in the task layer [18]. Decomposing a KBS in this way allows having several hierarchies of tasks where tasks can be mixed and matched and different task compositions can be built to solve various problems.

Interface Layer: User interface systems enable interactions of KBSs with users [14]. For efficient communication, these interactions should consist of two main aspects of (a) receiving inputs from users that outline users’ organization preferences, limitations or requirements. These inputs are used during the reasoning process to refine problem-solving strategies and achieve a dynamic and customized solution based on users’ needs; (b) representing the outputs of reasoning and task layer based on users’ criteria for selecting, filtering and grouping outputs.

4 Designed Knowledge-Based System Framework for Quantity Take off and Cost Estimation

4.1 Framework Overview

We have developed a KBS framework to provide a streamlined, 3D parametric model based quantity takeoff and cost estimation for construction products. This framework is represented in Figure 2 and includes the 4 layers of domain, reasoning, task and interface, designed for the precast concrete products which comprises the area chosen to implement a proof of concept for this research effort. This is an ongoing effort and so far the focus has been on developing a knowledge base and rule libraries. Several precast companies have collaborated and provided their company standards, practice manuals and their historical project cost estimation information. The principal researcher of this effort co-located for a few weeks with company experts to collect information

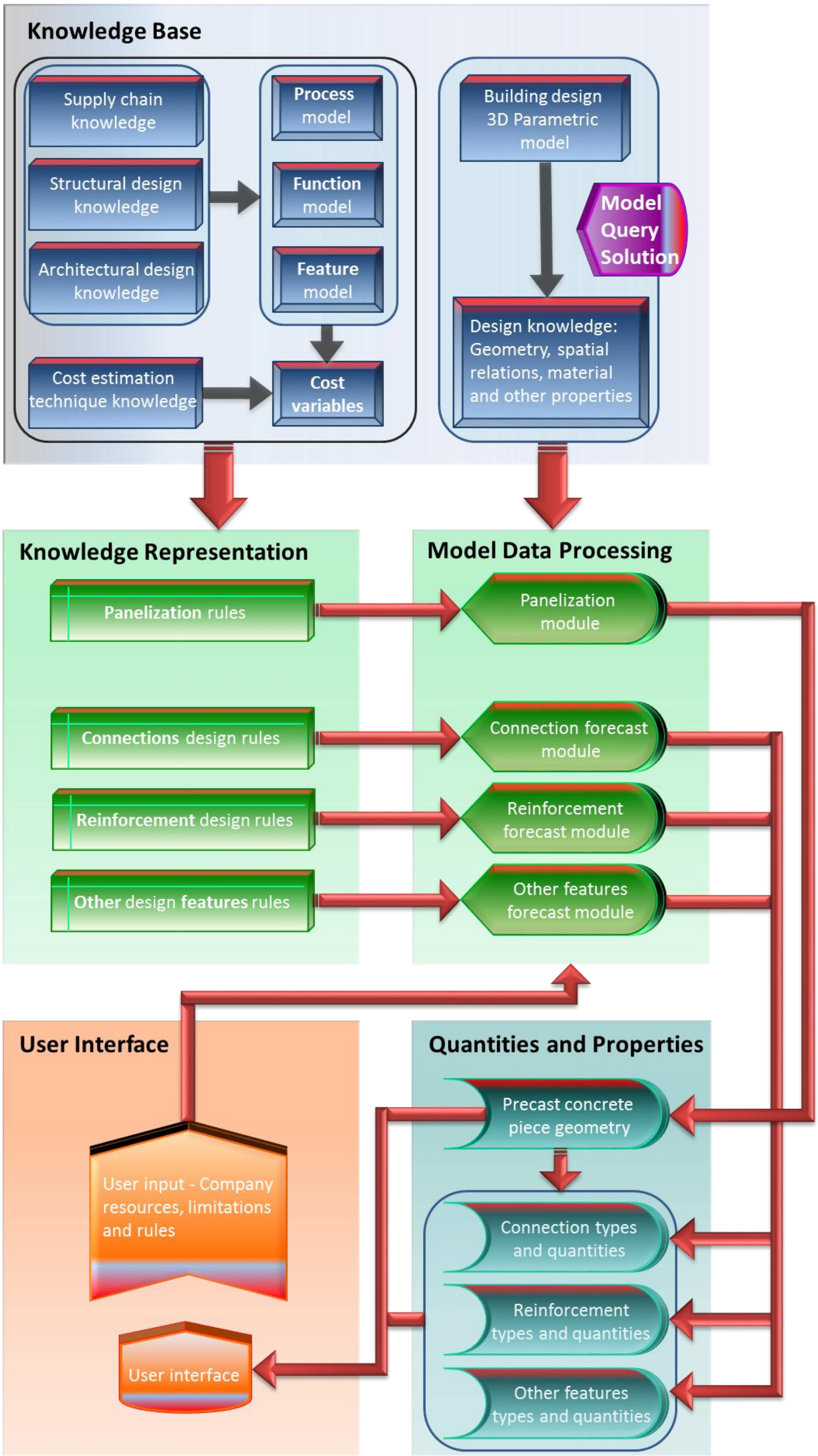

Figure 2. Developed framework for knowledge-based quantity takeoff and cost estimation

from estimators, structural engineers, plant managers and erectors; to observe their QTO and CE process; and to formulate the inference rules with the help of these experts. The knowledge base and reasoning rules are being developed both for architectural and structural precast concrete products.

The method involves the following steps:

- Studied different cost estimation conventions practiced in the precast concrete industry. Analyzed performance of different cost estimation methods and documented the results of the study in [19].
- Devised a combined feature- and function-based analytical cost estimation method [19] as the most suitable for the intended estimation level of detail and accuracy.
- Decomposed precast concrete products into their functional components and identified features required for each function.
- Developed a process map for quantity takeoff and cost estimation for each function and feature.
- Identified cost-driving attributes of each feature and specified the parameters required to measure the impact of each attribute on cost of a project. These variables comprise the information items necessary for precast concrete cost estimation.
- For the information items typically implicit design models or absent from models, defined the rules to infer knowledge about them and created a rule library for each function.

Expected results of implementation include enhancement of the design models to make them suitable for QTO and CE activities, extraction and representation of model information using the industry convention format and measurement units, and forecasting parameters of absent features by inferring new knowledge.

4.2 Domain Layer: Knowledge Base

The domains studied in order to develop an example knowledge base that guided the listed steps from product decomposition to process mapping and rule development included architectural and structural design, and supply chain analysis (fabrication, transportation and erection) of precast concrete products. The focus of knowledge acquisition was on those domain aspects that are interdependent with quantity take off and cost.

4.2.1 Cost estimation methods

Cost estimation methods can be categorized as intuitive and analogical methods used in early design stage CE and analytical methods for late design stages. A detailed study of these CE methods used in different project stages and analysis of the performance and shortcomings of each method is published by authors in a separate paper [19].

CE methods used in early stages of a project mostly can work only with a limited number of variables and provide a rough approximation of cost of a project suitable for budgeting. Considering this, they are not suitable for a more detailed CE process when there are more design information available and for instance geometry of building and different spaces within a building, type of building structure and location of structural elements are determined. Hence, for this research work, we use analytical CE method. Analytical methods decompose a product using operation-, tolerance-, feature-, and activity-based modeling.

We used a combination of activity- and feature-based product decomposition. We studied variety of design features that compose a specific product type, the supply chain process and activities that are required to fabricate each feature, and identified design variables that affect cost of each activity and therefore are important to be provided for cost estimators.

The main goal of the reviewed CE methods has been defining relationships of different design variables to cost of a project using historical data and applying various machine learning and optimization methods [20]. The focus in building the knowledge base of this framework is not to define cost relationships, rather to identify existence of those relationships between different variables and cost of a project and providing value of these variables to users, when they are not readily available in design models, through building a rule library and a rule processing engine. When the value of different variables are determined and provided to users they can then plug them in their formulas that are built based on their production process and local economic conditions.

4.3 Reasoning Layer: Rule Library and Inference Engine

As shown in Figure 2, we structured the reasoning layer by developing specific-purpose modularized rule libraries for various functions (e.g. connections, reinforcement, finishing, etc.) of different precast concrete product types (e.g. columns, beams, slabs, etc.). Rule libraries are being developed using different inference mechanisms to infer new knowledge for QTO and CE of different aspects of a product. These rules will be applied on the information extracted from 3D parametric design models as well as user inputs regarding company limitations and preferences. We will use a combination of generic inference tools many times found as off-the-shelf inference shells and specific purpose reasoning modules developed for domain applications.

These modules represent the rules and reasoning for three major purposes:

(i) Enhancing models using combination of implicit and explicit information to make them suitable for QTO and CE activities without the need to create new models from scratch: As explained earlier, while the unit of QTO and CE for precast concrete products is a precast concrete piece, in the architectural design models often pieces of precast concrete products are not distinguished. Through extracting the geometric and spatial relation information of products from models and applying panelization (modularization) rules developed based on the acquired domain knowledge, precast concrete model objects can be panelized to represent acceptable approximations of precast concrete pieces.

(ii) Extracting and representing explicit model information using the industry convention format and measurement units. This information mainly include dimension, surface area volume and weight measurements and properties like material.

(iii) Forecasting information about features absent from design models: Detailed design of many key features of precast concrete products like connections, reinforcement and form stripping and lifting inserts for the most part is performed by structural engineers who work for precast concrete contractors. The process is costly and time consuming and normally is performed after winning the bid and securing the project. During the QTO and CE activities information related to these features are mostly absent from models. Similar to the model enhancing process, relevant information for each feature is identified and extracted from design models. Rules to infer new knowledge about these features are developed which will forecast the value of cost-driving feature parameters (e.g. number and type of reinforcement elements).

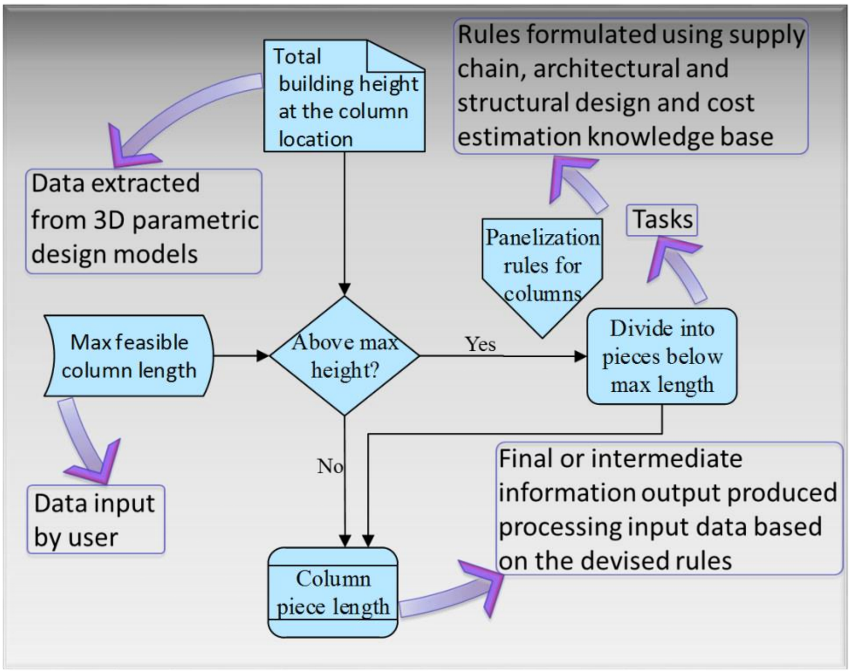

Figure 3. Example of a problem-solving method structure: inputs, outputs and actions

4.3 Problem Solving Methods and Knowledge Roles

Bases of the developed rules are PSMs and their knowledge roles. An example of application of a PSM and different knowledge roles is illustrated in Figure 3. In this example if the value "50'" is extracted from a design model as an observable for "total building height", it can be abstracted to "above max height" using another observable of "45'" which is a user input for "max feasible column height". This abstracted observable will be followed by applying a solution abstraction of “divide to pieces below max length”, which will produce the solution of “column piece length”.

Note that to generate this solution, the PSM requires another set of inputs which are “panelization rules for columns”. These rules are the end results of the process of knowledge acquisition from domain experts and knowledge representation. These rules themselves comprise of a cluster of PSMs that define the actions and the rationale behind each action and their implementation might require additional inputs.

This figure also shows how different classes of inputs including data, information and knowledge which are acquired from different sources including parametric design models, users and domain experts are used to carry out a task and infer new knowledge. The nature of these inputs also cover a wide range from dynamic (e.g. design model data that is project-specific and often even changes throughout a project lifecycle) to relatively static (e.g. panelization rules based on architectural, structural and supply chain rules) that can almost be considered fixed until standards or production technology change at which point they need to be refined.

5 Conclusion

This paper presented a framework developed for a streamlined knowledge-based quantity takeoff and cost estimation of precast concrete products using 3D parametric models. The focus was on development of a knowledge base and rule libraries for extracting implicit knowledge used by experts in CE. Validation will be inferred when estimators accept and have confidence in the results of CE from BIM models The next step, after completion of the knowledge base and rule libraries, is to implement this framework for selected categories of precast concrete products on one of the major domain software platforms.

References

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

[2] Forgues, D., Iordanova, I., Valdivesio, F., and Staub-French, S. Rethinking the cost estimating process through 5D BIM: A case study. Construction Research Congress 778-786, 2012.

[3] Akerkar, R. A. and Sajja Priti Srinivas. Knowledge-based systems. Sudbury, MA, USA: Jones & Bartlett Publishers, 2010.

[4] Dhaliwal, J. S., and Benbasat, I. The use and effects of knowledge based system explanations: theoretical foundations and a framework for empirical evaluation. Information Systems Research, 7(3), 342–362, 1996.

[5] Shehab, E. and Abdalla, H. An intelligent knowledge-based system for product cost modeling. International Journal of Advanced Manufacturing Technology, 19:49–65, 2002.

[6] Chan D. S. K. and Lewis W.P. The integration of manufacturing and cost information into the engineering design process, International Journal of Production Research, 38(17), 4413-4427, 2000.

[7] Bouaziz, Z., Younes, J. B. and Zghal, A. Cost estimation system of dies manufacturing based on the complex machining features, The International Journal of Advanced Manufacturing Technology, 28(3-4), 262-271, 2006.

[8] Ko, K. H., Pochiraju, K. and Manoochehri, S. An embedded system for knowledge-based cost evaluation of molded parts. Knowledge-Based Systems, 20, 291–299, 2007.

[9] Sobolewski, M. Federated P2P Services in CE Environments, Advances in Concurrent Engineering. A.A. Balkema Publishers, ISBN 90 5809 502 9,13–22, 2002.

[10] Koonce, D., Judd R., Sormaz D. and Masel, D. T. A hierarchical cost estimation tool. Computers in Industry 50: 293–302, 2003.

[11] Lee, S. K., Kim, K. R. and Yu, J. H. BIM and ontology-based approach for building cost estimation. Automation in Construction, 41, 96-105, 2013.

[12] Studer, R., Benjamins, V. R. and Fensel, D. Knowledge engineering: principles and methods. Data & knowledge engineering, 25(1), 161-197, 1998.

[13] R. Neches, R.E. Fikes, T. Finin, T.R. Gruber, T. Senator, and W.R. Swartout, Enabling technology for knowledge sharing, AI Magazine, 12(3) (1991) 36-56.

[14] Li, B. M., Xie, S. Q. and Xu, X. Recent development of knowledge-based systems, methods and tools for one-of-a-kind production. Knowledge-Based Systems, 24(7), 1108-1119, 2011.

[15] Fensel, D. (2002). Ontology-based knowledge management. Computer, 35(11), 56-59.

[16] Kim, C. H., Weston, R. H., Hodgson, A. and Lee,

K. H. The complementary use of IDEF and UML modelling approaches. Computers in Industry, 50(1), 35-56, 2003.

[17] Gruber, T. R. A translation approach to portable ontology specifications, Knowledge Acquisition, 5(2) 199-220, 1993.

[18] Schreiber, G., Wielinga, B. and Breuker, J. (Eds.). KADS: A principled approach to knowledge-based system development. Academic Press, 11, 1993.

[19] Aram, S., Eastman, C., Beetz, J. Qualitative and Quantitative Cost Estimation: A Methodology Analysis, ICCCBE 2014 & 2014 CIB W78, Orlando, Florida, 2014(Accepted).

[20] Niazi, A., Dai, J.S., Balabani, S., and Seneviratne, L. Product cost estimation: technique classification and methodology review. J. Manu. Sci. E-T ASME, 128 (2), 563–575, 2006.