# A FRAMEWORK AND PROCESS LIBRARY FOR HUMAN-ROBOT COLLABORATION IN CREATIVE DESIGN AND FABRICATION

A Dissertation
Presented to
The Academic Faculty

by

Shani Sharif

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Architecture, College of Design

Georgia Institute of Technology
August 2020

**A FRAMEWORK AND PROCESS LIBRARY FOR HUMAN-ROBOT COLLABORATION IN CREATIVE DESIGN AND FABRICATION**

Approved by:

Dr. Russell Gentry, Advisor
School of Architecture
*Georgia Institute of Technology*

Dr. Larry Sweet
School of Computer Science
*Georgia Institute of Technology and Amazon*

Dr. Dennis Shelden
School of Architecture
*Georgia Institute of Technology*

Dr. John Haymaker
Director of Research
*Perkins + Will*

Dr. Negar Kalantar
Department of Interior Design
*California College of the Arts*

Date Approved: May 15, 2020

To my amazing parents, sister and loving husband

# ACKNOWLEDGEMENTS

The accomplishment of this work could not have been possible without the support of my professors, Ph.D. peers, family, and friends. First, I would like to thank my advisor, Dr. Russell Gentry, for his continuous guidance, care, and encouragement throughout these years. I sincerely appreciate his tremendous support since my first day at Georgia Tech. He gave me the confidence to initiate and taught me to collaborate and share. I also would like to thank Professor Chuck Eastman, who has been a great mentor and invaluable inspiration to me. I am incredibly grateful that I had the chance to be part of his research team. His enthusiasm for research and intellectual rigor has been a great source of inspiration for me and my research work. Additionally, I would like to extend my sincere gratitude to Professor Larry Sweet, who unconditionally shared his wisdom and knowledge with me and supported me in this journey. He has become an excellent mentor and role model for me. I would like to express my gratitude to the other members of the doctoral committee, Dr. Dennis Shelden, Dr. John Haymaker, and Dr. Negar Kalantar, for their guidance and support through this process.

I consider myself extremely lucky to have such a loving and supporting family. I am genuinely grateful for the unconditional love, support, and constant encouragement of my mother, Zohreh, and my father, Shahriar, and my sweet sister, Rojia. Words cannot express how grateful I am to my beloved husband, Reza. None of these would have been possible without his continuous support and encouragement.

I would like to thank the faculty and staff in the College of Design, especially my academic advisor, Robin Tucker, for her exceptional care, passion, and support, and

Professor Scott Marble for his trust and providing me with the opportunity to teach. I am also grateful for the help and support of Jacob Tompkins, professor Tristan Al-Hadad, Dr. Thanos Economou, and professor Daniel Baerlecken.

I also am grateful to my amazing friends and colleagues at the Ph.D. programs in the Schools of Architecture and Computer Science at Georgia Tech, with whom I share many amazing memories and great conversations. Special thanks to Paula Gomez, Marcelo Bernal, Donghoon Yang, Andreas Cavieres, Francisco Valdes, Roya Rezaee, Khatereh Hadi, Kereshmeh Afsari, Varun Agrawal, Yong Cheol Lee, Matthew Swarts, Pedro Soza, Laura Florez, Sabri Gokmen, and Heather Ligler for their moral support and friendship through the years.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

## LIST OF SYMBOLS AND ABBREVIATIONS

| | |
|---|---|
| 2D | Two-Dimensional |
| 3D | Three-Dimensional |
| AEC | Architecture, Engineering, and Construction |
| AISC | American Institute of Steel Construction |
| B-rep | Boundary Representation |
| BFS | Breadth First Search |
| BIM-M | Building Information Modeling for Masonry Initiative |
| BIM | Building Information Modeling |
| BOM | Building Object Models |
| BPMN | Business Process Modeling Notation |
| BSI | British Standards Institute |
| CMU | Concrete Masonry Units |
| CNC | Computer Numerically Controlled |
| CSG | Constructive Solid Geometry |
| DFS | Depth First Search |
| ER | Entity Relationship |
| EER | Enhanced Entity Relationship |
| FEM | Finite Element Method |
| GFRP | Glass Fiber Reinforced Polymer |
| glulam | Glued-Laminated |
| GUID | Globally Unique Identifier |
| HSV | Hue, Saturation, and Value |
| KRC | KUKA Robot Controller |

| | |
|---|---|
| KRL | KUKA Robot Language |
| LOD | Levels of Development |
| MUD | Masonry Unit Database |
| OCCS | Omniclass Construction Classification System |
| RDBMS | Relational Database Management System |
| RGB | Red, Green, and Blue |
| RSI | Robot Sensor Interface |
| SKU | Stock Keeping Unit |
| SQL | Structured Query Language |
| TCP | Tool Center Point |
| UDP | User Datagram Protocol |
| UHPC | Ultra-High-Performance Cement |
| XML | Extensible Markup Language |

# SUMMARY

In the last two decades, the increasing affordability of industrial robots, along with the growing maturity of computational design software, has led architects to integrate robots into their design process. Robots have exceptional capabilities that enable the fabrication of geometrically complicated components and assembly of complex structures. However, the robot control and motion programming tools currently being adopted by designers were all initially intended for engineering-based manufacturing industries. When using computer-controlled tools, designers cannot adapt their designs to the production process in real time. Current industrial robot control systems force the designer to envision and embed all of the required machining data in the digital model before the fabrication process begins. This requirement makes the process of design to fabrication a unidirectional workflow.

In pursuit of a solution, a growing body of research is exploring various human-robot collaboration methods for architectural practices. However, many of these studies are project-based, targeting the ad hoc needs of a particular robotic application or fabrication process. Consequently, this dissertation investigates a generalizable framework for human-robot collaboration that is rooted in the principles of distributed cognition. As an essential part of the research argument, the role of the tools of production in the formation of a designer's cognitive system is considered.

This framework, defined for a bi-directional design and fabrication workflow, relies on and integrates material and fabrication feedback into the design process. The framework has three main components: interactive design, adaptive control, and a design and

fabrication library. While different aspects of these components have been studied to various extents by other researchers, this dissertation is the first to define them in an integrated manner. Next, the requirements for each of these elements are introduced and discussed in detail.

This dissertation focuses in more detail on the library component of the framework because compared to the first two components, it is the least investigated solution to date. A structure for the library is proposed so that the tacit knowledge of makers could be structured, captured, and reused. At its core, the library is a process-centric database where each process is supported by a set of tools, instructions, materials, and geometries required for the transformation of a part into its final form. Finally, this study demonstrates the generalizability of the library concept through a series of experiments developed for different material systems and with various robotic operations.

# CHAPTER 1. INTRODUCTION

Robots can be powerful and flexible fabrication tools. In the last two decades, the increasing affordability of industrial robots, along with the growing maturity of computational design software, has led architects to integrate robots and numerically-controlled fabrication machines into their design process. A growing body of research is exploring novel methods for incorporating industrial robots into the digital fabrication and assembly of building components (Brell-Çokcan and Braumann 2012, McGee and Ponce de Leon 2014, Reinhardt, Saunders, and Burry 2016, Willmann et al. 2018). The robot control and motion programming tools currently being adopted by designers were all initially intended for engineering-based manufacturing industries (Kolarevic 2004). This technology facilitates the industrial mass production of components with known properties and processes, and thus enables more predictable outcomes. However, engineers and designers (including architects) have different approaches to their design, experimentation, and production processes.

Engineering methods are based mostly on deductive reasoning, while design-based activities rely on abduction. Deductive reasoning derives conclusions via reasoning based on accepted facts and universal premises. This form of logic is self-contained, meaning that conclusions based on deductive approaches do not offer any new findings beyond the assumptions that underly the initial arguments. Conversely, abduction is a form of inference that makes probable conclusions from the information emerging from an investigation. Abduction leads to explanations or hypotheses based on observed phenomena or data, and in combination with prior experience (Merriam-Webster 2020, Kolko 2010). Deductive engineering processes focus on increasing the economy and repeatability of robotic operations, and thus facilitate standardization for the mass production of parts. In a deductive approach, the desired geometry is established a priori, and the materials, tool paths, and fixtures are selected to generate the desired geometry within the constraints of the system. Conversely, abductive design-based processes prioritize the customization of parts production, with an emphasis on creativity and uniqueness. Designers develop their concepts as a result of experimentation, focusing on the reciprocal exploration of form, testing of material behavior, and determination of robot kinematics.

This design-centric rationale that emphasizes both the aesthetic and functional aspects of production has a long history. The approach was at the center of traditional craftwork, the Bauhaus method in the early periods of architectural modernism (Celani 2012), and the current digital fabrication movement with its focus on fabrication methods and material properties as generative design factors (Menges 2012a). Specific characteristics distinguish digital fabrication from traditional craftwork. Digital fabrication allows for increased speed, scale, precision, and complexity, and makes possible the repetition and production of multiple instances of the same object. The combination of computer-aided design (CAD), computer-aided manufacturing (CAM) models, machining setups, and prepared materials allow for the process to produce not only the first instance of an artifact, but also the means to create subsequent copies. In a perfect case, there is one-to-one mapping between a complete CAD-CAM model, fabricated physical model, and actual building part or assembly. Additionally, designers are able to receive real material feedback because digital fabrication machines allow for prototyping with the materials of production (such as metal, concrete, or wood), often at full scale.

For both traditional craftwork and digital fabrication, the tools, materials, and means of production are essential parts of the design process. The main difference resides in the designer's handling of physical matter. Industrial robots and computerized, numerically-controlled (CNC) machines play an intermediary role in the process, creating a divide between the acts of designing and making. When using computer-controlled tools, designers cannot adapt their designs to the production process in real time (Gramazio, Kohler, and Oesterle 2010). Current industrial robot control systems require that the designer comprehensively understand the design object and embed detailed design and machining data into the digital model before the fabrication process begins. Consequently, the process of design to fabrication is mostly a unidirectional workflow in which the designer must predict the material state, tool selection, fixture positioning, and robot motion planning, usually based on prior experience.

## 1.1 Problem Definition

Recent research in the field of computational design has identified technological gaps resulting from unidirectional design-to-fabrication workflows. Researchers have highlighted the potential for humans and machines to act as complementary collaborators in the design-making process (Brugnaro, Figliola, and Dubor 2019, Garcia del Castillo Lopez 2019). The future of digital fabrication depends on a redefinition of the relationships between and interactions among human designers and machine fabricators. Human designers are the creative force in the process, and their vision is executed via specifications by machine fabricators designed to produce the designed artifacts. A successful human-robot interaction system relies on one key component of human intelligence: the ability to create tools to extend the brain's cognitive system. These tools help the brain complement its basic modes of processing and compensate for its weaker cognitive capabilities (Clark 2004). Examples of such weaker capabilities include the recall and execution of long sequences of operations such as those required to program and run industrial robots. Human-industrial robot collaboration depends on digital technology complementary to the cognitive capabilities of the human brain. A human-robot collaborative system can be successful if it leverages the strengths of the human cognitive system such as recognizing patterns, modeling simple world dynamics, and manipulating objects in the environment. At the same time, a successful collaborative system will compensate for human cognitive weaknesses by employing the inherent capabilities of robots and computers (Clark 2001, Hutchins 1995).

## 1.2 Research Question and Hypothesis

This dissertation investigates the elements required for an interactive human-robot collaboration framework to facilitate the creative design of production processes. Realizing a successful human-robot collaborative system requires an improved understanding of the nature of the human cognitive activities involved in the process of digital design-making. The factors necessary for the effective and comprehensive integration of design with making are illustrated through recently unearthed premises of cognitive science. New scientific approaches have defined cognition as a distributed processing system in which the brain, body, tools, materials, products, and social and material contexts are closely related to and interact with one another in cognitive activities (Malafouris 2004, Hollan, Hutchins, and Kirsh 2000, Hutchins 2000). This dissertation describes the distributed cognition theory and adopts its concepts for an analysis of the interaction between design conception and the act of production. As an essential part of this argument, the role of the tools of production in the formation of a designer's cognitive system is also discussed (Clark 2004, McCullough 1998, Norman 1998) and the requirements for the development of interactive robotic fabrication technologies addressed.

## 1.3 Overview of the Proposed Framework

This dissertation considers the role of feedback in the development of interactive fabrication systems. A human-robot collaborative system should address the different needs that emerge during various stages of the design concept formation, detailed design development, and production processes. In concept development, the ability of the robotic system to facilitate the designer’s design process and enhance their overall creativity is of highest importance. In the detailed design and production stages, building more accurate design prototypes and avoiding potential fabrication errors are essential. This dissertation asserts that there exist three main components of a bi-directional workflow in the human-robot interactive process: (1) interactive design, (2) adaptive control, and (3) a design and fabrication library (see Figure 1-1). The requirements for each and current efforts toward their development are introduced below and discussed extensively in the body of the dissertation.

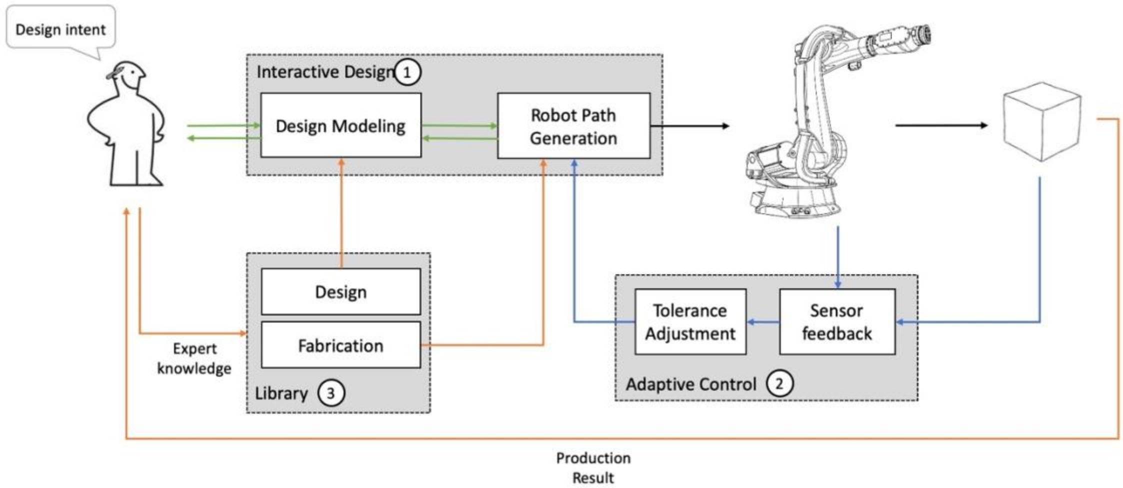


**Figure 1-1. Proposed bi-directional design-fabrication workflow.**

The first element of this proposed framework is interactive design, which provides flexible and intuitive robot control and programming tools for non-roboticist designers. Recently developed interactive software tools often incorporate and build upon graphical programming editors such as Grasshopper for Rhinoceros 3D or Autodesk Dynamo. By using applications such as KUKA|prc and HAL, designers can program and simulate industrial robots directly from parametric modeling environments during the conceptual and detailed design stages.

The second piece of the framework is adaptive control, which provides robotic solutions for dealing with the unpredictability associated with the environment and materials, and inaccuracies in the fabrication process. This element relies on the simultaneous or reciprocal collaboration of human designers and robot fabricators as complementary partners. The main requirement for such a collaborative model is a system for robot movement based on real-time fabrication data obtained from various sensors. Adaptive control assists designers with making informed decisions and using program processes that integrate uncertainty as a part of fabrication, rather than being required to envision and incorporate all of the steps ahead of time in a CAM model.

The last component of the suggested framework is the library, a knowledge database of design and fabrication methods and practices. The library is envisioned as a means of assisting designers with the decisionmaking process by offering design options and robotic production techniques based on the design, tool, and material intent. At its core, the library is a process-centric database of objects, functions, and paradigms for working with robots in the context of a design environment. The entire database model is structured around (fabrication) processes as the core element of the library. The other required and supporting parts of the system are tools, materials, geometries, and instructions. All supporting data are arranged around and feed into each fabrication process.

### 1.4 Methodology

This dissertation addresses the human-robot collaboration framework in two different stages. First, by considering the three requirements of the system – interactive design, adaptive control, and the library – in an integrated manner, the completeness of such a system is demonstrated relative to other work that focuses only on a single component. Second, a more in-depth examination of each element is conducted.

Interactive design is an element that has been investigated extensively by other researchers, and a few solutions are used widely by the design community. Thus, this dissertation includes a literature review on this topic but does not focus on developing solutions for interactive design. The second component is adaptive control. Studying new tools and methods for adaptive control has become the epicenter of recent robotic fabrication research. This dissertation investigates this topic by developing a technique for integrating KUKA RSI (a robot sensor interface) into the Grasshopper graphical programming environment. Adaptive control is a software-hardware intensive module that requires expertise in sensing and image processing. While this dissertation addresses this topic, it does not focus deeply on this module because many other studies are currently exploring the same problem. Instead, this dissertation focuses primarily on the library component of the framework because compared to the two first components, it is the least investigated solution to date.

In this dissertation, the overall process-centric structure of the library is introduced. Then, various elements required for the library, including materials, instructions, and processes, are investigated to identify the critical aspects of a robotic fabrication process for architectural applications. The generalizability of the library concept is demonstrated through a series of experiments developed in different material systems and with various robotic operations. The main goal of this portion of the dissertation is to identify commonalities in the processes and methods so that the requirements of the library can be adequately described.

### 1.5 Structure of the Dissertation

The remainder of this chapter outlines the dissertation and describes how each chapter contributes to building the case for a human-robot collaboration framework. Each of these chapters is based on one or more journal articles or conference proceedings published by the author and her colleagues. Citations to these articles are provided here and at the beginning of each subsequent chapter in the dissertation.

**Chapter 1 Introduction** This chapter gives an overview of the dissertation and introduces the study. It articulates the problem and formulates the research question. Finally, it provides an overview of the proposed framework and research methodology.

**Chapter 2 Shifting Design Cognition from Craftsman to Digital Maker** This chapter defines the dissertation's core problem, investigates the underlying requirements for a successful solution, and proposes a framework for overcoming this challenge. For this purpose, the chapter examines the intermediary role of digital fabrication machines in changing the discourse of design cognition as it relates to the action of making. It emphasizes the development of conditions for completing the connections among mind, body, and the technological environment as realized through interactive participation in the process of design-making, shaping human-machine interactions in a manner that unifies the design and fabrication processes.

The citation to the journal paper is as follows:

Sharif, Shani, and Russell Gentry. 2020. "Shifting Design Cognition from Craftsman to Digital Maker." *Under review at the Journal of Design Studies*.

An earlier version of this study was published in the following conference proceeding:

Sharif, Shani, and Russell Gentry. 2015. "Design Cognition Shift from Craftsman to Digital Maker." 20th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2015), Daegu, Korea.

**Chapter 3 Adaptive Industrial Robot Control for Designers** This chapter investigates adaptive control, which is the second component of the human-robot collaboration framework. This work presents a system that allows designers to adaptively control an industrial robot via a 3D modeling environment, producing real-time feedback with respect to visual images of the object and the robot's pose during the fabrication process. The chapter shows the qualitative results of a fabrication process in order to demonstrate how the proposed system improves interaction and collaboration between designers and robots, in contrast to the iterative process commonly followed.

The citation to the conference paper is as follows:

Sharif, Shani, Varun Agrawal, and Larry Sweet. 2017. "Adaptive Industrial Robot Control for Designers." ShoCK - Proceedings of the 35th eCAADe Conference, University of Rome, Rome, Italy.

**Chapter 4 Robotic Design and Fabrication Library** This chapter discusses the core features of the third component of the human-robot collaboration framework. The process-centric data model for the library is explained, as well as supporting modules including materials, tools, geometry, and instructions.

**Chapter 5 Parametric Product Libraries: Development and Implementation of a Masonry Unit Database** This chapter discusses the requirements for defining and storing the material properties for a library, focusing on masonry units as an example. The work provides a description of the development of an industry standard data model for concrete and clay masonry units. It begins with a review of classification systems and building object model libraries. The chapter also explains a process for the parametric representation and storage of digital masonry units that allows for speedy retrieval from a database. By using masonry as an example, this research shows how complex building objects that have heretofore been outside the scope of BIM models and processes can be integrated into modern design and construction procedures.

The citation to the journal article is as follows:

Sharif, Shani, and Russell Gentry. 2020. "Parametric Product Libraries: Development and Implementation of a Masonry Unit Database." *Under review at the Journal of Automation in Construction*.

An earlier version of this study was published in the following conference proceedings:

Sharif, Shani, and Russell Gentry. 2015. "BIM for Masonry: Development of BIM Plugins for the Masonry Unit Database." Real Time - Proceedings of the 33rd Education and Research in Computer Aided Architectural Design (eCAADe) Conference, Vienna, Austria 16-18 September 2015.

Sharif, Shani, Russell Gentry, Chuck Eastman, and Jeff Elder. 2015. "Masonry Unit Database Development for BIM-Masonry." 12th North American Masonry Conference, Denver, Colorado.

Gentry, Russell, Shani Sharif, Andres Cavieres, and David Biggs. 2016. "BIM schema for masonry units and walls." Proceedings of the 16th International Brick and Block Masonry Conference.

**Chapter 6 Reinforced Composite Brick Masonry for Offsite Robotic Fabrication** This chapter investigates the processes and instructions for robotic fabrication and libraries, focusing on reinforced masonry as an example. The work discusses an algorithmic solution for testing the constructability of various complex wall forms for a robotically fabricated masonry system. It introduces a new composite construction technique for reinforced masonry that serves as an alternative to traditional wet or dry stacking. It also demonstrates that by using a set of developed rules, designers can test their brick assemblies with curved surface forms to ensure structural stability and robotic constructability.

The citation for the conference paper is as follows:

Sharif, Shani, Daniel Griffin, and Russell Gentry. 2020. "Reinforced Composite Brick Masonry for Offsite Robotic Fabrication." Submitted to RobArch 2020 Conference, under review.

**Chapter 7 Robotic Sheet Metal Folding: Tool vs. Material Programming** This chapter focuses on the fabrication process aspect of the library. The work explores how deductive engineering thinking, as opposed to an abductive design rationale, can influence how robotic methods for fabricating building components are developed. By discussing the advantages and disadvantages of each approach, this research argues that both engineering and design should be considered necessary and complementary processes in the development of new creative fabrication solutions. For this study, robotic dieless sheet metal folding was chosen and investigated as the primary fabrication technique.

The citation for the journal paper is as follows:

Sharif, Shani, and Russell Gentry. 2020. "Robotic Sheet Metal Folding: Tool vs. Material Programming." *Under review at the Journal of Automation in Construction*.

**Chapter 8 Discussion and Conclusion** This chapter summarizes the results of the dissertation research in response to the research question and describes its contribution to the greater body of knowledge and state of the practice. It also discusses the generalizability and limitations of the roadmap presented for human-robot collaboration efforts. The chapter concludes by recommending areas of future research on this topic.

# CHAPTER 2. SHIFTING DESIGN COGNITION FROM CRAFTSMAN TO DIGITAL MAKER

**Abstract**

The process of design and fabrication involves complex cognitive activity in which the human brain is part of a more extensive cognitive system that encompasses the brain, body, tools, materials, and environment. Cognition resides in the interaction of all these elements at different stages of design and making. This research investigates the intermediary role of digital fabrication machines in changing the discourse of design cognition as it relates to the action of making, inquiring into the path diverging from traditional craftwork. This study emphasizes the development of conditions for completing the connections among mind, body, and technological environment as realized through interactive participation in the process of design-making, shaping human-machine interaction in a manner that unifies the design and fabrication process.

## 2.1 Introduction

Fabrication-focused architectural design processes, like any other creative practice, involve cognitive activities that develop innovative and meaningful products. The design-centric rationale that emphasizes both the aesthetic and functional aspects of production has a long history. This approach was at the center of traditional craftwork, the Bauhaus method in the early periods of architectural modernism (Celani 2012), and the current digital fabrication movement with its focus on fabrication methods and material properties as generative and integrating design factors (Menges 2012a). While for both traditional craftwork and digital fabrication the tools, materials, and means of production are essential parts of the design process, the main difference resides in the designer's handling of physical matter. Industrial robots and computer numerically-controlled (CNC) machines play an intermediary role in the process, creating a divide between the acts of designing and making. In using computer-controlled tools, designers can no longer adapt their designs to the production process in real time. As Gramazio and Kohler have argued, "achieving a sophisticated building component … can be compared to methods used by manufacturers from pre-industrialized ages. Despite the similarities, today the action of material handling is indirect through the use of CNC machines as opposed to the instant feedback about the work in progress the skilled manufacturer received through the tool in his hand. With CAM, the tool is controlled through explicit routing data, which leaves no room for interpretation and adaptation" (2010).

Recent research in the field of computational design has highlighted the potential for human and machine to act as complementary collaborators in the design-making process. Some of these studies have proposed the development of new interfaces for integration of physical and digital environments, augmenting actuators with sensors to incorporate real-time feedback on the material's state during the production process (Brugnaro, Figliola, and Dubor 2019, Betti et al. 2018, Brugnaro and Hanna 2017, Raspall, Amtsberg, and Peters 2014, Johns, Kilian, and Foley 2014, Mueller, Lopes, and Baudisch 2012). Realizing this goal requires a better understanding of the nature of the human cognitive activities involved in the process of digital design-making. Therefore, this research enquires into the intellectual processes of a maker, and specifically the path diverging from traditional craftwork (Keller and Keller 1993, O'Connor 2006, Malafouris 2008) and leading to the act of digital making.

The factors required for the effective and comprehensive integration of design with making can be illustrated through recently unearthed premises of cognitive science. New scientific approaches have defined cognition as a distributed processing system in which the brain, body, tools, materials, products, and social and material contexts are closely related and interact with one another in any cognitive activity (Malafouris 2004). This research outlines this distributed cognition theory and adopts its concepts for an analysis of the interaction between design conception and the act of production. These concepts are first discussed in the realm of craftwork, and then expanded into digital design and fabrication. As an essential part of this argument, the role of the tools of production in the formation of a designer's cognitive system is also discussed (Clark 2004, McCullough 1998, Norman 1998), and the requirements for the development of interactive robotic fabrication technologies studied. This work concludes with a discussion of feedback in the development of interactive fabrication systems.

To illustrate these cognitive concepts, this research draws examples from projects developed as part of three graduate and undergraduate courses: "Construction Materials, Systems, and Fabrications," "Introduction to Robotic Fabrication," and "Robotic Fabricates." These courses were developed by and offered at the Digital Fabrication Laboratory in the School of Architecture at the Georgia Institute of Technology.

## 2.2 Distributed Cognition

Following the development of artificial intelligence concepts in the 1960s, the dominant view in the study of human cognition involved conceptualizing the brain and mind as an analogy for computer hardware and programs, structures disembodied and detached from the material world (Malafouris 2004). However, a new premise of cognitive science focuses on the development of human cognition in the embodied world. In this view, human cognition is greater than the activities that happen inside an individual's brain. The material environment is part of the cognitive system, and as Hutchins has described, "cognitive activity is sometimes situated in the material world in such a way that the environment is a computational medium" (Hutchins 2000). Different theories in this school of cognitive science address a variety of aspects of this issue. Key theories include embodied cognition, as formulated by George Lakoff and Mark Johnson; situated cognition, as conceived of by Jean Lave, Lucy Suchman, and Etinne Wagner; distributed cognition, as developed by Edwin Hutchins; mediated cognition, as theorized by Lev Vygotsky; and extended cognition, as orchestrated by Andy Clark (Malafouris 2004, Nersessian 2008).

Distributed cognition theory provides a plausible explanation for an analysis of the cognitive process from design to making. The distributed cognition framework describes the mechanisms involved in the cognitive processes beyond those that happen inside the brain of the individual. This theory emphasizes that all cognitive activities, including design and fabrication, are in close relation to embodied action, representing a dynamic coupling of mind and matter (Hollan, Hutchins, and Kirsh 2000, Malafouris 2004). Human cognition is not limited to the mind as a passive representational system, as has been portrayed by AI-influenced cognitive scientists. The cognitive system encompasses complex coordination between internal (i.e., memory, attention, executive function) and external (i.e., material and environmental) structures (Hollan, Hutchins, and Kirsh 2000). The brain's role is to act as a mediating component in a complex network of processes that continuously "loop" among the brain, body, and surrounding environment (Clark 2001). Based on this framework, this research argues that the tools and materials of fabrication are an integral part of the design education and creation process (Sharif 2013).

In recent decades, a growing number of makers has shifted from using human-controlled tools in traditional production systems to CNC machines in digitally-driven production systems. However, regardless of the medium of production, the structural elements of the cognitive activity remain the same. In both practices, the brain, body, tools, materials, products, and social and material contexts are in close collaboration with one another. Schön (1992) described the design process as a reflective conversation with the materials of the design situation. Section 2.3 discusses the interdependent relationships among the brain, body, and act of production as viewed through the lens of manual craftwork. This argument is complemented in Section 2.4 by an assessment of the main impacts of the shift from human-controlled to numerically-controlled tools in the cognitive process.

## 2.3 Manual Fabrication

In artful craftwork, the craftsman requires the knowledge and skill necessary for making purposeful objects with their hands. “Craft is habitual skilled practice with particular tools, materials, or media, for the purpose of making increasingly well-executed artefacts. Craft is the application of personal knowledge to the giving of form”

(McCullough 1998). Craftwork involves bodily activity, the use of hand-controlled tools, and direct manipulation of materials. The craftsman is in control of the tools, whether they are hand-held such as chisels and pliers, or more complex and mechanical like milling machines. The result of the work is a unique artifact, the quality of which is highly dependent on the skill level of the craftsman.

### *2.3.1 Cognitive Function: Knowledge and Action in Tandem*

In a study of the production of crafts, Keller and Keller (1993, 1996) described an inherent dynamic interrelationship between the knowledge of design and act of production. Craftwork is an interaction between the internal representation of the craft object and action of making as external representation. In this view, knowledge and action are integrated, simultaneously prerequisite, and consequences of each other. Knowledge has both social and material aspects. It includes the internal image of the designed object, goal of production, and conceptualization of the production sequence (Brereton 2004). The craftsman's knowledge is selectively derived from prior production experiences and ideas about the world. Relevant past experiences are prioritized based on the closest correspondence with the current design situation.

Craftwork is a dynamic process. The act of making, material object produced, and ongoing perception of this action transforms and enriches the designer's prior organization of knowledge. This continuous evolution is a critical part of the process, as the designer's knowledge is never adequately detailed and sufficiently precise to predict all possible situations and outcomes of the craftsman's action throughout the process of production.

The craftsman can also never fully predict all required actions because they are affected by the materials, tools, and environmental conditions at each moment (O’Connor 2006).

In each task, the craftsman must consider different criteria for production, including functional adequacy, the aesthetics of the design, and various techniques and procedures that involve tools and machines, financial constraints, and material conditions. As Keller and Keller argued, "these dimensions operate as positive forces for action [and] not determinants of outcome" (1993). To establish a new production plan, the craftsman builds upon conceptualizations, actions, and operations of similar previous production tasks that serve as either successful or defective feedback. However, this pre-conception only initializes the task; the design concept evolves concurrently with the craftsman's act of production and feedback received from the evaluation of the materials and objective conditions of the work. The co-evolution of design conception and artifact production is influenced by under-constrained tools and material conditions that allow for creative development.

In manual work, the initial conception of a task provides only general guidance for its undertaking. This plan is not required to be comprehensive. Based on the craftsman's evaluation of the result and required revisions of their decisions, necessary details can be incorporated into the task during each step of the process. If limitations in the required skillset or tools inhibit the process of making, the craftsman can extemporaneously modify the design objectives to fit the new requirements and fix mistakes, ultimately improving the final product's quality. In short, the ongoing development of the conceptual task is the source and outcome of the materialization of the craftsman's actions (see Figure 2-1).

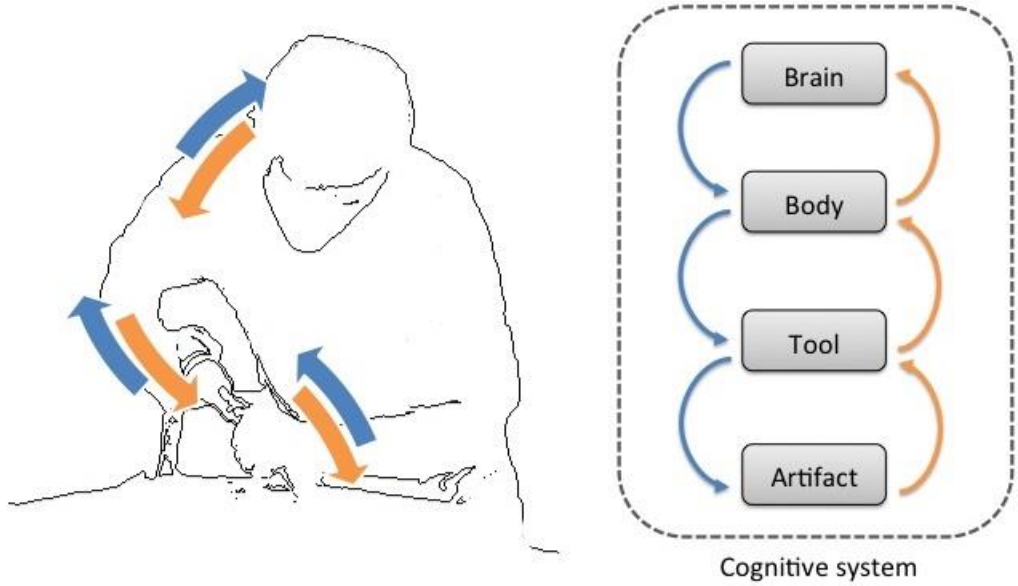


**Figure 2-1: Representation of the craftsman's cognitive system.**

*2.3.2 Study: Active Feedback*

The cognitive process of human-controlled tools in traditional production systems was observed as part of the exercises assigned in the course "Construction Materials, Systems, and Fabrications." Course projects were designed to introduce students to material properties, fabrication processes, and tools and machinery appropriate for working with four major classes of building construction materials: wood, concrete, metal, and polymers. The goal of the course was to foster a practical assessment of the complexities associated with the integration of multiple material systems, selection criteria for a given fabrication method, and sequential logic required for assembly (Valdes, Cavieres, and Gentry 2013).

The final class project was a collective effort to design, fabricate, and assemble a panelized wall system comprised of tile panels, connection joints, and columns (see Figure 2-2). This exercise had two stages: fabrication and assembly. In the first stage, students worked independently to make three tiles and connector elements. Each part had to be produced with only one material and a single fabrication technique (either by hand- or computer-controlled machinery). In the second stage, students collaborated on connecting and assembling the individual components into a final wall system. They were encouraged to envision the final assembly while designing and fabricating their components. Specifically, they had to take into account the tolerances required to accommodate small variations and inconsistencies in the final dimensions and connection positions of the parts.

**Figure 2-2. Wall panel with tiles made of hardwood, thermoformed plastic, and concrete. Tiles were connected to glulam wood columns with metal joints.**

One of the class projects, the design and fabrication of a glued-laminated (glulam) wood column as part of the final wall assembly, provided a useful case study for observing the distributed cognition concept in practice. The main goal of this exercise was to make a part that would act as the main structure of the wall system and support all of the infill panels. The glulam columns were fabricated using hand-controlled woodworking machinery including a planer, jointer, table saw, and chop saw, in addition to other essential hand-controlled tools such as mallets, wood clamps, and chisels. Although students cannot be considered skilled artisans, their fabrication process still represents the connected flow of bodily activity, use of hand-controlled tools, and direct manipulation of materials. Mainly, their process for functional criteria gathering, design, and making through interacting with woodworking tools exhibited the main attributes of distributed cognition because their knowledge (i.e., design learning) and actions (i.e., manual skill learning) were engaged in tandem.

There were only two main specifications. Each column had to be made according to exact cross-section and length dimensions and made with six plies of ½-inch wood to achieve a total width of 3 inches, and the adjacent layers of wood could not have aligned butt joints. These instructions formed the basis of the design and fabrication requirements. Students were asked to start with an initial shop drawing of their conception of the final product that represented each piece with dimensions and permissible tolerances. They then prepared these drawings based on the requirements.

In the next step, students chose their raw wood pieces to begin the fabrication process. Soon they found that the available stock material in the lab would not necessarily fit with their envisioned dimensions for each wooden piece. Thus, they had to modify their initial shop drawings based on their evaluation of the material conditions. During the making process, they often had to adjust the thickness or length of each wooden piece in real time. There were numerous mistakes and inaccuracies encountered in achieving the desired dimensions, due to students' lack of skill and familiarity with manual woodworking techniques and tools. They had to adjust the sizes of the new wooden pieces to compensate for incorrect dimensions in the parts they'd cut in previous steps. In making these minor adjustments, students could control the final size of the assembled column and make sure it satisfied the permissible tolerance range. For example, if incorrect settings for the planer machine caused an individual ply to be cut ¼" narrower than the desired ½" thickness, the student had to cut the next piece ¼" thicker to achieve the total thickness required at the end of the process. Although none of the students had significant prior woodworking experience, most were able to modify their initial designs based on real-time feedback from the conditions of the material and their ability to control their tools (see Figure 2-3).

The students' process of cognition followed the pattern illustrated in Figure 2-1. The feedstock and tooling were readjusted and coordinated in real time to achieve the end product. The shop drawings and design specifications could not accommodate unpredicted dimensions in the wood supply or students' inefficiency in operating the analog machines. In many cases, the end products, though successful, deviated significantly from the conceptual ideas and illustrations in the initial design documentation.

**Figure 2-3. A glued-laminated wood column in production (left) and the final product (right).**

### 2.4 Digital Fabrication

Advances in digital technology and, specifically, digital fabrication machinery and software tools have created the opportunity for the integration of design, analysis, manufacturing, and assembly in architecture, as well as the chance to reassess and potentially redefine the relationship between design and production (Kolarevic 2004). As Oxman and Oxman (2010) have emphasized, digital "fabrication is a revolution in the making of architecture" that "has a profound influence upon the definition of the requisite knowledge base of the architect." Fablabs in architecture schools have become testbeds for the development and assessment of new design and production knowledge (Gershenfeld 2005, Celani 2012, Blikstein 2013). As discussed above, one method of understanding this requisite knowledge is inquiring into the cognitive resources that mark the shift from traditional craftsman to digital maker. Consequently, investigating the cognitive synergy of knowledge and action in the digital design-production process calls for clarification of what we mean by digital fabrication and why it has become a trendy approach in design education.

By utilizing digital fabrication technologies, architects can quickly materialize, test, and evaluate the constructability and efficiency of their designs for building components and assemblies. These new tools provide novel techniques for the production of both commonly used and emerging construction materials. Specific characteristics distinguish digital fabrication from traditional craftwork, as discussed below:

1. In a perfect case, there is one-to-one mapping between a complete CAD-CAM model, fabricated physical model, and anticipated actual building part. The digital model and physical prototype have almost all the details, components, and features of the final product, with all its complexities and intricacies (see Figure 2-4).
2. Digital fabrication machines allow for prototyping with the materials of production (such as metal, concrete, or wood), often at full scale, as opposed to common model materials like chipboard or foam. This feature provides the designer with material feedback regarding strengths and weaknesses of the design concept and manufacturing process (Valdes, Cavieres, and Gentry 2013). Additionally, digital fabrication preserves a material’s logic. Designers can use the material’s properties as factors in generating geometric form (Menges 2012b).
3. Digital fabrication allows for increases in speed, scale, precision, and complexity, and provides the opportunity for repetition and production of multiple instances of the same object.
4. In the design and fabrication of building parts, connections are key, either connections between parts made of the same material or parts with different materials or material combinations. The careful design of these connections based on the affordances and limitations of machines, fabrication methods, and material

properties after considering their tolerances can allow for seamless attachment in the final assembly.

5. In the process of design, the designer must have an a priori perception, often naive, of how the actual final product will be made. This perception is based on the designer's experience or projection, as enhanced by empowering computer modeling tools. The design concept can be tested and improved through the development of physical models.

**Figure 2-4: One-to-one mapping between the 3D model and fabricated parts. From left to right: 3D CAD model, CNC milled wooden module, cast concrete module with CNC milled foam mold, and thermoformed plastic module with CNC milled wood composite.**

### *2.4.1 Cognitive Functions: Missing Feedback*

The industrial robots, CNC tools, and related simulation and CAM systems currently adopted by designers were all initially developed for engineering industries (Kolarevic 2004). These technologies were developed for the industrial mass production of components with known problems and processes and predictable outcomes (Field 2015 ).

Because these fabrication machines are created for industrial applications, they enforce upon their users a very constrained set of interactions and experiences. The process is mostly a one-directional workflow, starting with designing the model in the CAD environment, and then preparing it and robot toolpaths based on the geometric data and machining settings, generating machine-readable code, transferring the data to the robots or CNC tools, setting up the material on the machine or in the robot cell, and finally running the program and executing the production of the final part. Although there are some levels of interactivity in the initial stages of CAD/CAM model development, after initiation of the machining process, the designer has no control over fabrication (see Figure 2-5). While these robotic software and hardware packages perform adequately and effectively in production manufacturing, they do not provide room for concurrent interactive, creative, or exploratory design-to-fabrication activity.

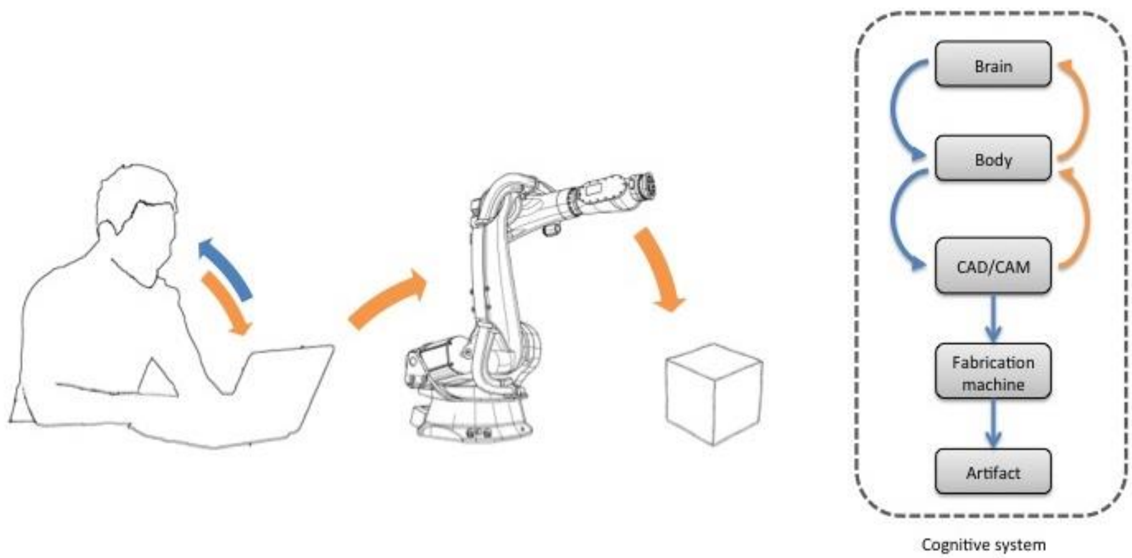


**Figure 2-5: Cognitive system of a digital maker.**

From an embodied and distributed cognition point of view, fabrication tools are used not only as an instrument of physical production, but also for creative thinking and design. In other words, as Clark explained: "the brain's role is crucial and special, but it is not the whole story. In fact, the true power and beauty of the brain's role was that it acted as a mediating factor in a wide variety of complex and iterated processes, which continually looped between brain, body, and technological environment, and it is this larger system that solved the problem" (2001). This cognition loop is complete in manual fabrication (see Figure 2-1).

In the manual fabrication process, the craftsman initiates the design task only with a pre-conception of the design object and process; this is permissible due to the under-constrained fabrication settings and environment. However, the digital maker is limited to only a partially connected system by the constraints of the industrial robots and CNC machines. This workflow forces the designer to have a comprehensive view of the design object from the beginning and embed all design details and machining data in the digital model before the start of the fabrication phase. The design concept affects the process of machining, but no feedback during the action of making can affect design ideation or the organization and content of the design model.

### *2.4.2 Study: Missing Feedback*

We examined students' cognitive process of working with machine-controlled tools – specifically, a six-axis robotic arm – as part of the course "Introduction to Robotic Fabrication." The main goal of this course was to introduce students to the fundamentals of robot kinematics and fabrication methods for architectural production. For course projects, students used Rhinoceros, a 3D CAD software, and Grasshopper, a visual programming environment, to create and model their designs. The main fabrication tool was a Kuka robotic arm, KR Quantec Pro (KR 120 R2500), with a payload of 120 kg and arm reach of 120 cm. In order to make end effectors for the robot and prepare other material parts for fabrication, students had access to other machines and tools in the fabrication lab including a waterjet cutter, laser cutter, three-axis CNC machine, and manual wood and metalworking tools.

Here, we highlight the fabrication processes for two projects from three semesters: (1) a hotwire foam-cut wall panel and (2) a folded sheet metal hanging canopy. In each project, students began with a design concept for which they created a 3D digital model in Rhino3D. Next, they used the Kuka|PRC plugin for Grasshopper to create a robot toolpath to materialize their design idea. Then, the source code for robot control was exported and transferred to the robot controller (see Figure 2-6Figure 2-8). However, when students initiated the fabrication process, they faced major problems that were different from those encountered during the manual fabrication process. During the manual fabrication process (see Section 2.3.2), students could modify their design and adjust to the conditions of the material product. In the digital fabrication process, they could not correctly identify or prevent errors during the course of the robot's operation. Students faced many failed attempts that demanded troubleshooting and modification before they achieved a satisfactory result (see Figure 2-7Figure 2-10). The majority of problems were due to incorrect materials placement, material variations, unanticipated material behavior, unpredicted robot singularities, and automatic robot path optimization. These errors caused waste in terms of both time and material because the entire process had to be repeated after required editing of the robot toolpath's simulation file.

For the first project, students were asked to design and build a self-standing wall panel out of polystyrene foam blocks by using a hotwire foam cutter attached as an end effector to the robot arm. One of the 3D modeled designs and robot toolpath are illustrated in Figure 2-6. After these steps, students began the fabrication process, but as discussed above, they faced many failed attempts (see Figure 2-7). The cut units at the end of each trial were completely different from what they had modeled in the digital environment. However, since the students had no tactile control or any other type of precise feedback during the robot's operation phase, troubleshooting was a major challenge.

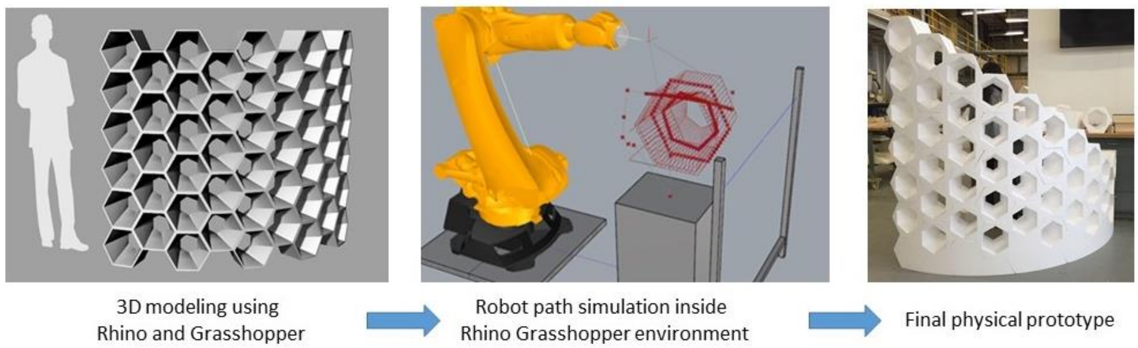


**Figure 2-6. Geometric modeling and robot path planning and simulation in the Grasshopper visual programming environment.**

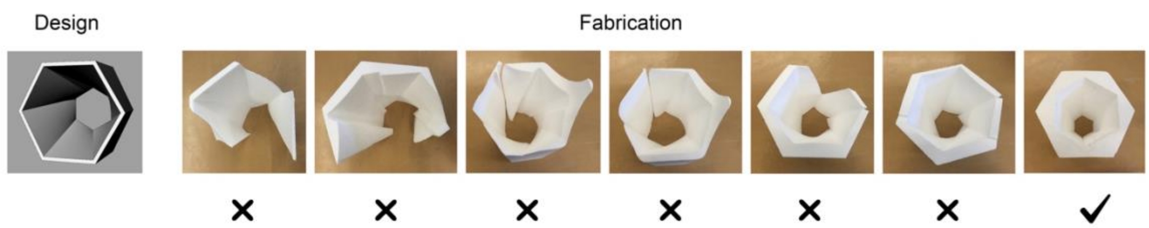


**Figure 2-7. Failed attempts at fabrication of the module before successful results. Failures were due to incorrect material placement, robot position singularities, and undesirable robot path optimization.**

After much trial and error, the following major problems in the setup process were identified:

- Wire temperature setting: The wire was heated via electrical resistance. If the wire temperature was not set up correctly, the heat from the wire would not vaporize the material immediately before contact.

- Speed of wire movement: If the robot end effector moved too fast, the heated wire would not have sufficient time to vaporize the material, and instead the wire would come in direct contact with the foam, causing friction.
- Toolpath settings: The robot toolpath settings had to be properly adjusted for the desired outcome. By default, many of the robot's movements were optimized for the shortest path along a Bezier curve, while the cut pattern required an interpolated point curve.
- Material positioning: In certain positions, six-axis robot arms ran into singularities. This is when the robot can reach a certain arm joint position in an infinite number of ways, causing movement failure.

For successful outcomes, designers must predict and implement the correct material state, tool selection, fixture position, and robot motion plan, usually based on prior experience. All of these factors are known to an experienced fabricator. However, for novice practitioners or those who seek new creative fabrication methods, there are always high cost and time penalties for rework before the desired design and fabrication intent are achieved. The same process pattern was observed in another project that focused on robotic sheet metal folding.

In recent decades, there have been numerous studies on the industrial applications of robotic sheet metal folding techniques (Liao and Wang 2003, Lavallee, Vroman, and Keshet 2011, Aomura and Koguchi 2002). In this project, students investigated a new a folding technique for metal that could achieve multiple folds on one module with as few robot movements as possible. The method would take advantage of the inherent ductility of the sheet metal by following a fold sequence based on a connected series of crease lines.

This concept originated from a simple twisting of a strip of material, relying on the fact that material reduction along crease lines creates areas with lower yield points than the rest of the material. The hypothesis was that multiple folds on a module could be achieved through a twisting motion (see Figure 2-9). Unlike other common practices for sheet metal folding, the force direction was not perpendicular to individual fold lines, but rather followed the transition path between the 2D geometric state and final 3D state of the module.

**Figure 2-8. Geometric 3D model and fabricated prototype of a robot-based sheet metal twist folding project.**

Students designed a series of model configurations to test their hypotheses. Initially, the flat cut parts were twisted manually using two grippers, but the action applied the indented logic of the robotic folding process (see Figure 2-9, middle). Students then identified the most suitable relationship between the final geometry and fold pattern and established the fabrication logic for the part. They then tested different cut patterns with variable factors including the number of fold lines, angle between each set of fold lines, and angles of the fold lines relative to the strip. Next, the results of the manual study were repeated using a KUKA robot. However, students found that they could not replicate the same result using the robot. The product was an unevenly folded strip that in most test cases bent only at the center of the part. Students then needed to perform more tests to identify the main difference in the process, lowering the speed and implementing a stepwise rotation (see Figure 2-9).

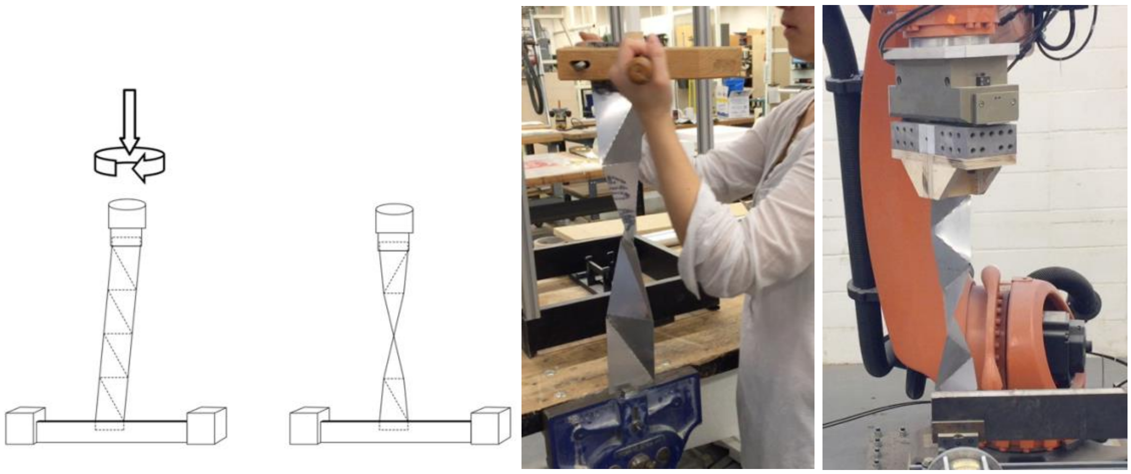

**Figure 2-9. Concept design for the twist-folding of sheet metal (left) and manual and robotic twist-folding tests (middle and Right).**

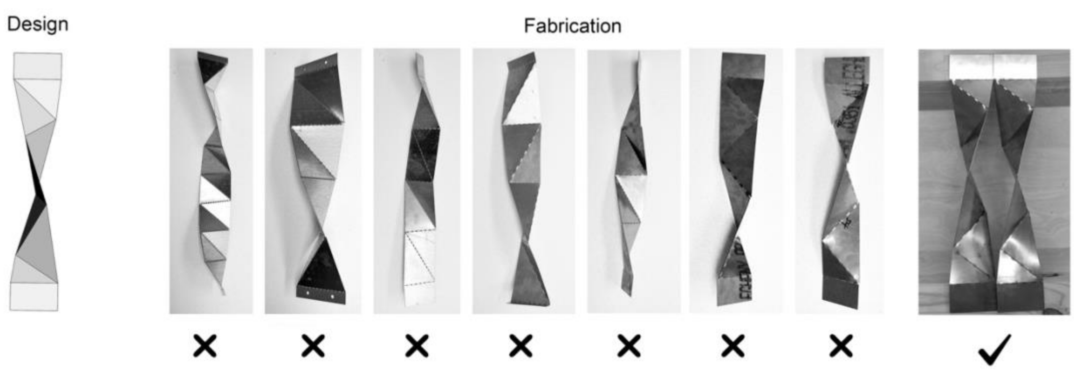


**Figure 2-10. Failed attempts at the fabrication of folded metal modules before the final successful result.**

In the case of the laminated wood column (see Section 2.3.2), students began with a limited set of materials. When errors emerged, they could find solutions to modify the fabrication method and still achieve the design intent. The final prototypes were quite different from the original specifications, at least with regards to the details, but the final outcomes were still considered successes. The manual fabrication experience was very different from the two digitally realized prototypes discussed here. The artifacts made with robots were exactly as generated by the robot codes. If students realized that the defined specifications were incorrect, they had to discard the damaged material, troubleshoot, modify their code, and begin the process again. The language of the digital tool (e.g., KRL, KUKA robot language) was an inseparable part of the production process and thus had to be considered as part of the cognitive process of the digital designer.

## 2.5 Closing the Loop

In the last two decades, research in both engineering and design has identified technological gaps resulting from one-directional design-to-fabrication workflow. Thus, the future of digital fabrication depends on a redefinition of the relationship and interactions between human designers and machine fabricators. As discussed in Section 2.2, one key component of human intelligence is the ability to create tools to extend the cognitive system, complement basic modes of processing, and compensate for weaker cognitive capabilities (Clark 2004). Human-industrial robot collaboration is a new workflow enabling digital technologies to perform as operations complementary to those of the cognitive capabilities of the human brain. A human-robot collaborative system can be successful if it can take advantage of the strengths of the human cognitive system, while also compensating for its weaknesses by leveraging the inherent capabilities of robotic and computer systems.

The human brain is particularly capable at tasks such as pattern recognition, the extraction and encoding of relational properties such as relative distance and parallelism, shape-based object identification and segmentation, and the modeling of simple world dynamics. However, human cognition is weaker in other tasks such as recalling and executing long or arbitrary sequences of operations (Clark 2004, Norman 1998). To maximize the benefits of a successful design and fabrication system, human and robot strengths should be utilized to the fullest (see Table 2-1).

**Table 2-1. Human Perception vs. Computer Perception, Weaknesses, and Strengths (Halim 2009, Clark 2004, Norman 1998)**

| Human perception | Robotic and computer systems |
|---|---|
| **Strength** | **Strength** |
| Creative design based on design intent | Accurate form generation with numerical inputs (parametric form development) |
| Alternative form development | Speed in numerical computation |
| Alternative fabrication technique development | Speed in physical operation |
| Synthetization and interpolation | Accuracy in operation (computational and physical), complex sequences of operations |
| Detection of trends, patterns, or anomalies in visual data (digital and physical design models) | Endurance and reliability |
| Learning from limited test cases | Consistency |
| Collaboration and integration (people, tools, senses) | |
| **Weakness** | **Weakness** |
| Unable to detect minor deviations in digital and physical prototypes | No creativity: problems synthesizing new rules |
| Easily tired of repetitive operations | Unable to produce alternate and "out of the box" solutions to compensate for inadequate design solutions |
| Biased and inconsistent | No common sense |
| Erroneous | |

In general, any creative design process consists of three main stages: conceptual design, detailed design, and final design production. In the conceptual design stage, there are many factors that must be decided by the designer, including the aesthetics of the design, functional adequacy, materials selection and condition, production techniques and procedures and tools, and financial and time constraints. At this stage of the design, the impact of design decisions is very high (Wang et al. 2002). Such choices are made based on the design intent and matched selectively with the designer's prior production experiences, based on the closest correspondence with the current design situation. The act of making, the physical prototype produced, and the designer's ongoing perception of this action transforms and enriches the designer's prior organization of knowledge. However, the designer's knowledge of a specific process is never adequately detailed and precise enough to predict all possible situations and outcomes of the production process. Thus, the design conception and artefact production co-evolve (see Section 2.4.2), metal folding study). The final design concept generated at this stage affects the product's shape geometry, materials selected for production, construction and manufacturing productivity, and the final product's quality. In the subsequent detailed design phase, it is difficult to compensate for or correct unanticipated aspects or shortcomings of the design concept developed at the conceptual design stage. Consequently, there is a high payoff for improving early knowledge integration and computational support for early decisionmaking (Cavieres, Gentry, and Al-Haddad 2011, Wang et al. 2002).

A human-robot collaborative system should address the different needs that emerge during the design concept and detailed design development and production stages. In concept development, the ability of the robotic system to facilitate the designer's design process and overall creativity is of highest importance. In the detailed design and production stages, the system's ability to build more accurate design prototypes and avoid potential fabrication errors is essential. Figure 2-11 illustrates three main components for a complete bi-directional workflow in a human-robot interactive process: (1) interactive design, (2) adaptive control, and (3) a design and fabrication library. The requirements for each of these elements and current efforts toward accomplishing them are discussed below.

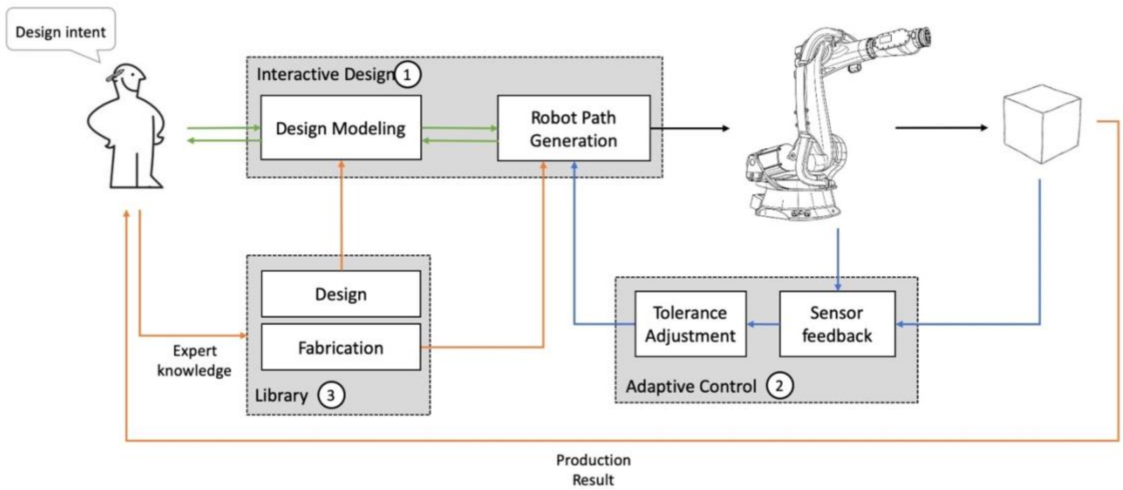


**Figure 2-11. Proposed bi-directional design-fabrication workflow.**

### *2.5.1 Interactive Design*

One of the integral parts of a complete bi-directional design-to-fabrication workflow is interactive design solutions (see Figure 2-11). Researchers in the field of computational design have focused on improving the design-to-robotic-fabrication workflow by developing more flexible and intuitive robot control and programming tools for non-roboticist designers. These new software tools incorporate graphical programming editors such as Grasshopper for Rhinoceros 3D or Autodesk Dynamo. Different plug-ins such as KUKA|prc (Braumann and Brell-Cokcan 2011), HAL (Schwartz 2013), and Scorpion (Elashry and Glynn 2014) for programming and the kinematic simulation of industrial robots such as KUKA, ABB, and Universal Robots have been developed for graphical programming editors such as Grasshopper3D and Dynamo BIM. These plug-ins provide architectural designers with the option to program and simulate industrial robots directly out of the parametric modeling environment, based on the geometric parameters of their designs. These tools provide interactive design and robot programming/simulation in the initial stages of the process.

However, most of these tools result in static robot control data files that must be transferred from a personal PC equipped with the robot programming tool to the robot's computer. After generating the robot's path and exporting the file, the connection between the design model and robot's movement path is completely severed. By the time the fabrication process begins, the designer has no control over the process in real time. In this process, the final physical prototype outcomes are still highly dependent on the predictive capability of the architect's design, fabrication, analytical, and process models (see Figure 2-12).

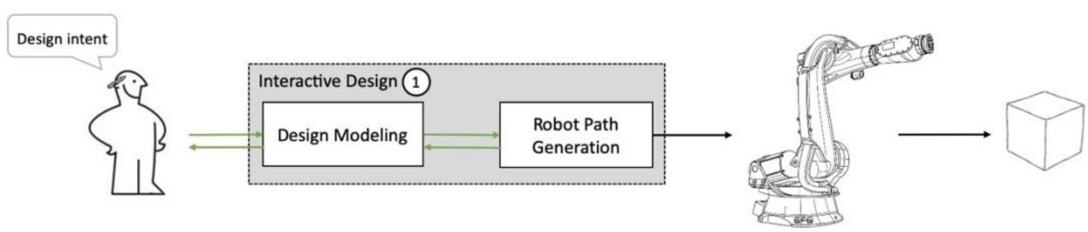


**Figure 2-12. Interactive design for robotic fabrication.**

### *2.5.2 Adoptive Control*

Main challenges for a robotic fabrication system include the unpredictability of the environment and material and inaccuracies in the fabrication process. These problems are especially important in complex situations requiring integration or assembly of complex parts and multiple materials. To solve this type of issue, researchers continue to develop the human-robot collaboration process, a system in which human and machine act as complimentary partners in a cooperative endeavor (Gramazio, Kohler, and Oesterle 2010, Nikolaidis et al. 2013, Field 2015 ). The main requirement for such a collaborative model is an adaptive control system for robot movement based on real-time fabrication data obtained from a sensor system (or a combination of 3D scanner, camera, depth sensor, force/torque sensor, etc.) (see Figure 2-13). By equipping robots with sensor systems, we can create platforms that assist fabricators with making informed decisions. Fabricators can then program a process that integrates uncertainty as a part of fabrication, rather than being required to envision and incorporate all of the steps ahead of time in a CAM model (Nikolaidis et al. 2013, Boerkoel Jr and Shah 2013).

Research in the field of human-robot collaboration in fabrication can be categorized into three main areas: (1) real-time robot control technology for industrial robots, (2) autonomous robot fabrication with real-time data, and (3) human-robot collaborative fabrication.

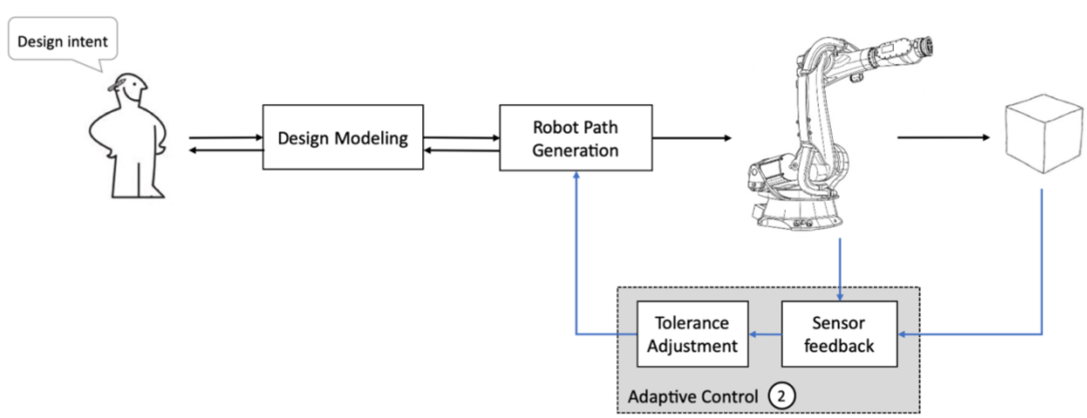


**Figure 2-13. Adaptive control for robotic fabrication.**

#### 2.5.2.1 Real-time robot control

Industrial robots are intended for manufacturing applications that require substantial safety features and predictive behavior. Consequently, manufacturers tend to make these robots closed systems, as opposed to more research-focused and flexible machines (Sharif, Agrawal, and Sweet 2017). Industrial robots are programmed with manufacturers' priority programming languages, such as KRL for KUKA robots. Their programming is usually performed offline via an external computer, with software tools such as KUKA.Sim Pro for KUKA or RobotStudio for ABB robots. While these tools provide features adequate for industrial applications, they have limited capabilities for adaptive control; there is no support for advanced mathematical operations or third party libraries to extend the primary robot language with user-defined methods or objects. In addition, in order to employ external input devices such as sensors, it is necessary to install supplementary software packages like Kuka.RobotSensorInterface (RSI) or KUKA.PLC mxAutomation.

Various research groups have investigated the creation of custom communication interfaces to act as middleware between user programs (such as CAD software tools in architecture) and robot controllers to make the interface of industrial robots more suitable for research and design fabrication applications. OpenShowVar is a Java-based cross-platform program for connecting KUKA controllers to remote computers (Sanfilippo et al. 2015). Another application, KUKA prc with mxAutomation integration, provides a plugin for Grasshopper 3D for real-time communication with KUKA industrial robots (Munz, Braumann, and Brell-Cokcan 2016). Another group of researchers has presented a means of integrating Kuka.RobotSensorInterface with Grasshopper3D (Sharif, Agrawal, and

Sweet 2017). Most recently, Machina.NET has emerged, a library for the programming and control of industrial robots that targets users in creative fields (e.g., designers and artists). This library provides a comprehensive solution for real-time robot control suitable for controlling systems that require concurrent responsiveness to sensory or user input (García del Castillo y López 2019, Garcia del Castillo Lopez 2019).

#### 2.5.2.2 Adaptive robotic fabrication

Robotic fabrication for architectural construction differs from that of the manufacturing industry as it has to deal with unpredictability in the environmental and material conditions. Adaptive control is mainly intended for detailed fabrication and construction. Robots can be programed to integrate uncertainty into the fabrication process. Autonomous robots receive raw real-time environmental and material data as input, analyze the information and act based on predefined conditional algorithms. Such algorithms can be used for either corrective path planning or conditional decisionmaking. Based on the sensor data, the gap between scan points and the desired shape is calculated. The adaptive control loop then adjusts the robot planning vector parameters to achieve the fabrication result closest to the design intent defined in the computer model. Such algorithms are used to fine tune the fabrication of each single part, based on acceptable tolerances.

Feng et al. discussed their computer vision-based pose estimation system that uses a single camera and visual marker-based metrology to detect and autonomously assemble non-unique and randomly placed building components such as bricks and blocks in pre-designed modular structures (Feng et al. 2015). In another study, researchers Investigated a similar topic, autonomous robotic assembly in an unstructured environment. However, the main difference in that study was that they used a 3D model of building components represented in a BIM model to detect objects via scanning without needing to sort or label them (Dawod and Hanna 2019).

#### 2.5.2.3 Human-robot collaborative fabrication

The last collection of research on this topic explores applications of human-robot collaborative efforts for creative fabrication. The main requirement of this type of model is a system that can capture and reflect feedback from both human and robot collaborators. Feedback is required to create a complete closed-loop interactive system encompassing the human designer, digital design environment, robot fabricator, and final product.

In a project called "Interlacing," Dorfler et al. presented a system for concurrent human and robot acts of placing wooden sticks such that the overall fabricated structure maintained structural integrity. In this project, the general shape of the structure was spontaneously controlled by the designer, while the robot checked the stability of the structure. The robotic system was able to receive and execute human commands in real time, instead of executing a set of predefined procedures (Dörfler, Rist, and Rust 2013). In another study, Brugnaro and colleagues performed various investigations of the concept of "digital craftsmanship." Their work focused on creating robotic systems that incorporated uncertainty in the fabrication process in order to take advantage of unanticipated formal configurations or performance behavior. By using sensor feedback and machine learning procedures, these researchers analyzed manual craftwork (see Section 2.3.1) and tried to provide a robotic system that would incorporate the coevolution of design by human and robot fabricators (Brugnaro et al. 2016, Brugnaro and Hanna 2017, Brugnaro, Figliola, and Dubor 2019).

### *2.5.3 Library*

The final element enhancing robotic fabrication is a component to assist with the decisionmaking process by offering a means of evaluating the adequacy of design choices, detailing, and materials selection for select robotic production techniques. With a given tool path, fixture location, and material property, the final shape of a fabricated part can be determined through simulation or experimentation. However, the main challenge a designer faces is inverse planning (i.e., going from a desired end shape to a tool path and fixture location) (Ponthot and Kleinermann 2006). In this type of scenario, a library can assist the designer by providing a knowledge database of methods and best design and fabrication practices. However, a digital fabrication library can only store and work with explicit knowledge, while most human fabrication skills are tacit in nature.

Knowledge of design can be categorized into two types: explicit and tacit (Eraut 2000, Nonaka and Konno 2005). While explicit knowledge can be represented in words and numbers and communicated systematically through data, scientific formulae, specifications, and manuals, tacit knowledge constitutes informal personal skills and is difficult to express, formalize, and share with others (Reber 1989, Bernal 2016a). Both tacit and explicit knowledge have important roles in the design-making process. Explicit knowledge of design is necessary for the direct exchange of ideas as shared through descriptions, written instructions, tool demonstrations, user manuals, and instructions encoded into drive robots. Tacit knowledge is more personal and gained through observation, induction, and participation (rather than formal inquiry). While a maker's knowledge in working with a material has a tacit nature, for computer-controlled tools this knowledge has to be translated into explicit information so that robots can receive the information, process it, and feed it back it into the system.

Efforts to integrate tacit knowledge into the process of digital design and fabrication can benefit from the concept of "history-enriched digital objects" proposed by James Hollan (Hollan, Bederson, and Helfman 1997, Hollan, Hutchins, and Kirsh 2000). Based on this theory, physical objects can record histories of use in ways that inform tasks and facilitate interaction. Hollan argued that these mechanisms can be replicated for digital objects and processes, "exploiting computation to develop a new history of interaction mechanisms that dynamically change to reflect the requirements of different tasks" (Hollan, Hutchins, and Kirsh 2000). Part of experts' tacit knowledge is using the material and spatial properties of the world to perform (analog) computations to cue recall, accelerating identification and action. Learning from demonstration is one technique that captures tacit knowledge, converting it into explicit knowledge that can be stored in a library and used for robot action planning. In this approach, robots learn to perform skills or specific tasks based on human expert actions (Nikolaidis et al. 2013, Brugnaro and Hanna 2017, Payne 2011). A design and fabrication library can be developed incrementally based on both experimentation and human expert knowledge. Both real and virtual experimentation on interpolating the results of these experiments could yield a complete series of options for different design choices. The database would then grow over time with data obtained from continued experimentation by multiple users.

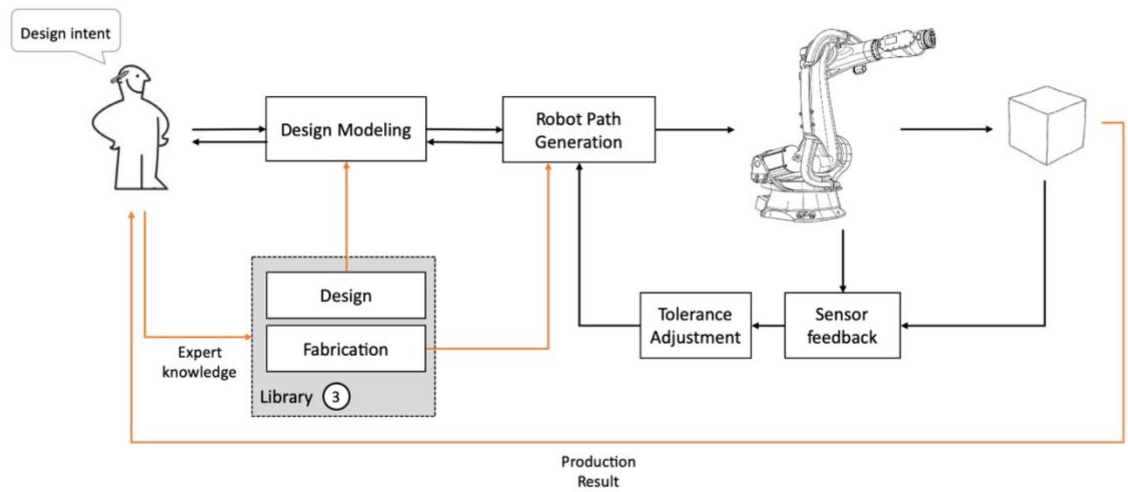


**Figure 2-14. A design and fabrication library for human-robot collaboration.**

## 2.6 Conclusion

This research examined the cognitive shift from traditional craftwork to the digital fabrication production process in light of distributed cognition theory, which studies cognitive activities beyond the brains of individuals and encompasses both the material and environmental context. In this definition of cognition, tools become extensions of human brains and bodies; they are more efficient and integrated with the cognitive system when used without conscious thought, focusing the attention of the user on the material. Thus, digital fabrication tools must be involved in the interaction between knowledge and embodied action. The computer-controlled machines of the future will be equipped with adaptive and interactive systems able to cooperate with human designers' "plastic brains" (Clark 2004), actively engaging in the process of design and making.

# CHAPTER 3. ADAPTIVE INDUSTRIAL ROBOT CONTROL FOR DESIGNERS


## Abstract

In this research, we present a system that allows designers to adaptively control an industrial robot via a 3D modeling environment, producing real-time feedback with respect to visual images of the object and the robot's pose during the fabrication process. Our work uses KUKA industrial robots due to their programmability and capability with regards to fabrication; we also employ Rhino 3D modeling software with the Grasshopper plug-in, which allows for visual programming. A Microsoft Kinect sensor is used to provide visual feedback during the fabrication process. In this work, we also present the methodology used to develop the system, explaining various design and architecture choices that allow for easy use while also ensuring that the system is open to further extension. We show qualitative results of the fabrication process in order to demonstrate how our proposed system improves interaction and collaboration between designer and robot, in contrast to the iterative process commonly followed.


## 3.1 Introduction

This research proposes a framework for enhancing the collaboration process between human designers and industrial robots. Recent technological advancements have led to the development of a new generation of industrial robots. Compared to earlier iterations, these robots are much more affordable, accurate, and flexible multi-purpose manipulators. All of these aspects make them optimal tools for creative and mass-customized architectural fabrication processes.

In the last decade, architects have adopted industrial robots for additive, subtractive, and deformative fabrication activities. However, robot control and motion programming software packages were originally developed for the industrial mass production of components in factories. Robot path planning and programming is engineered for specific production processes and predictable outcomes. The main criteria for these systems are functional accuracy and repeatability over an extended operational time period. Developed and confirmed robot toolpaths run for several months to secure the mass production of particular products. Conversely, creative design and fabrication processes in architecture are highly exploratory, meaning that architects must solve complexities related to new forms, materials, and fabrication mechanisms and unpredictable construction environments. Solutions for these creative endeavors rely on a reciprocal investigation of conceptual digital and material forms. Consequently, for activities that use digital design and fabrication, it is essential that tools facilitate a reciprocal design and fabrication development process.

However, using existing industrial robot control systems for architectural fabrication requires that the designer have a comprehensive view of the fabrication and machining process and embed all required considerations in the digital model before the start of the fabrication phase (Sharif and Gentry 2015b). This is a one-directional workflow (see Figure 3-1) in which the designer must predict the material state, tool selection, fixture positioning, and robot motion planning, usually based on prior experience (see Figure 3-2). This established workflow, a method that performs adequately and often effectively in production manufacturing, does not provide room for interactive creative design/fabrication activity. There is a high cost and time penalty for re-work.

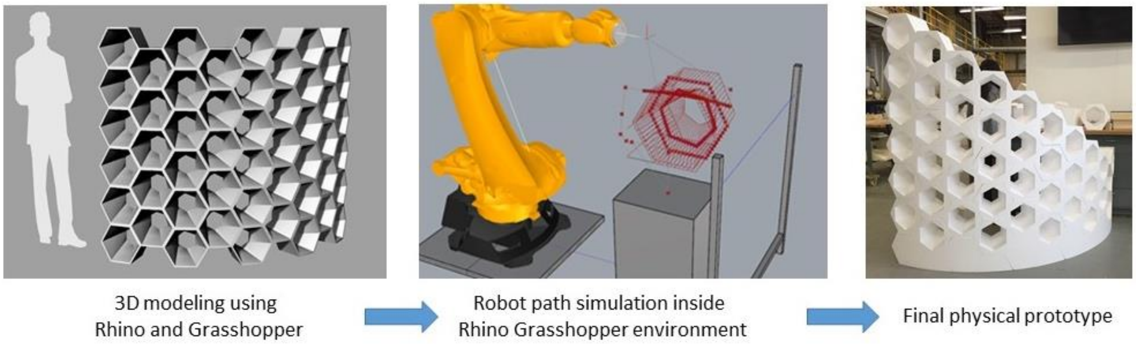


**Figure 3-1. Geometric modeling and robot path planning and simulation in a visual programming environment.**

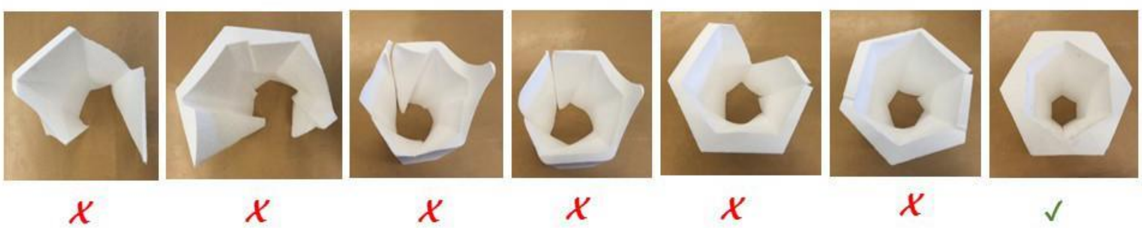

**Figure 3-2. Failed attempts at fabrication of the module before the final successful results; problems stemmed from incorrect materials placement, singularities, and unanticipated robot path optimization.**

As the conventional methods of robot control and motion programming were not developed based on the needs and skills of designers, researchers in these field have focused on producing more flexible and intuitive robot control and programming tools. These new software instruments have incorporated graphical programming editors such as Grasshopper. Different plug-ins like Kuka|prc (Braumann and Brell-Cokcan 2011), HAL (Schwartz 2013), and Scorpion (Elashry and Glynn 2014) for programming and kinematic simulation of industrial robots such as KUKA, ABB, and Universal Robots have been developed for Grasshopper and Dynamo. Plug-ins for graphical robot programming provide the option for architectural designers to program and simulate industrial robots directly out of the parametric modelling environment, based on the geometric parameters of their designs. These tools provide interactive design and robot programming/simulation in the initial stages of the process. However, most of these tools result in a static robot control datafile that has to be transferred from a personal PC equipped with the robot programming tool to the robot's computer. After path generation and export of the file, the connection between the design model and robot movement path is completely disconnected. By the start of the fabrication process, the designer has no control in real time. The final physical prototype outcomes are still highly dependent on the architect's predictive ability to design, fabricate, analyze, and process models.

Consequently, this research proposes an integrated framework that transforms the current one-directional workflow of the design-to-fabrication process into a comprehensive closed-loop methodology. By using the proposed system, designers can now monitor the fabrication process and control robots in real time. The system provides the functionality necessary for users to modify programmed robot toolpaths in response to materials tolerances and dynamic or unstructured environmental conditions that vary from the expected state. Changes to a robot's toolpath may also be required based on the designer's choice of a different course of action during the fabrication process in pursuit of a more desirable design and fabrication outcome.

## 3.2 Background

Robot manufacturers produce robots for two major target user types, manufacturing professionals and research and development practitioners. While the mechanical and hardware requirements of robots for these two user groups are similar, the software and control requirements are quite different. While the manufacturing industry requires more substantial safety features and easier operational interfaces, researchers, including design researchers, prefer greater control over the robot. In the last few years, manufacturers of robot have invested substantially in the development of research robots, also known as collaborative robots.

Collaborative robotics is a new research and development approach being pursued in the robotics industry. By design, collaborative robots are equipped with communications interfaces that provide accessible external control. These new interfaces and internal sensors (e.g., motor torques, joint torque sensors) and external sensor systems (e.g., cameras, 3D or laser scanners) enable the use of accessible programming to define robot toolpaths. Robots such as LBR iwaa from Kuka, Yumi from ABB, UR10 and UR5 from Universal Robots, and Baxter and Saywer from Rethink Robotics were developed based on the principals of human-robot interaction. They are usually lightweight and desktop-sized, and thus much more portable than their industrial counterparts. A major feature is that such robots can perform in close proximity to and in collaboration with human workers (Shepherd and Buchstab 2014).

The controller kernels of these robots are modular with open interfaces that enable the object-oriented programming of complex robot kinematics. They also make external control and integration of sensor systems relatively straightforward. Taking advantage of these integrated features, researchers have begun developing interactive robotic fabrication applications. Elshary and Glynn created a robot control plug-in for Universal Robots in the Grasshopper environment. Scorpion takes advantage of Java and Python object-oriented programming languages and their existing libraries for programming  and real-time visualization of robot toolpaths and configurations (Elashry and Glynn 2014). Other researchers have developed a system for onsite robot programming that leverages the embedded force torque sensors of the Kuka LBR iiwa robot to facilitate human-robot collaboration and manage materials tolerances during the fabrication process (Stumm et al. 2016).

While collaborative robots offer researchers excellent features, they suffer from lower payloads. A robot's payload is the maximum weight it can pick up or manipulate; the measure is especially important for architectural fabrication processes, as robots often must handle heavy construction materials such as concrete, metal, and stone and apply substantial force for processes such as milling, lifting, and assembly. While collaborative robots have about 5 to 10 Kg payloads, industrial robots offering medium payloads from 50 to 150 Kg and high payloads from 250 to 600 Kg that are more suitable for architectural construction processes. However, as discussed above, these robots are not equipped with flexible programming and control interfaces. In order to make high payload industrial robots suitable for research and design fabrication applications, it is necessary to develop custom communication interfaces that provide the flexibility required for adequate control.

### 3.3 Adaptive Control Interfaces for Industrial Robots

Kuka industrial robots are programmed in Kuka's own programming language, Kuka Robot Language (KRL) (2003). KRL is a text-based language that contains tool and machine movement commands, as well as data type declarations, conditional clauses, and interaction with tools and sensors via digital or analog input/output operations. KRL programming is usually performed offline via an external computer through software tools such as KUKA.Sim Pro or Kuka|PRC, along with a visual programming interface. While KRL's features are adequate for industrial applications, it has limited capabilities with regards to adaptive control; there is no support for advanced mathematical operations or third party libraries included to extend it with user-defined methods or objects (Sanfilippo et al. 2015). In addition, in order to use external input devices such as sensors, it is necessary to install supplementary software packages such as Kuka.RobotSensorInterface or Kuka.Ethernet KRL XML.

Various research groups have attempted to create custom communication interfaces to act as middleware between user programs (e.g., CAD software in architecture) and robot controllers in order to make the interface of this type of industrial robot suitable for research and design fabrication applications. One open-source communication interface for Kuka industrial robots, OpenShowVar, is a Java-based cross-platform package developed by researchers at Aalesund University College in Norway (Sanfilippo et al. 2015). JOpenShowVar is a custom-designed program that connects to a Kuka controller from a remote computer via TCP/IP, without using a Kuka software package such as Robot Sensor Interface (RSI) or Ethernet.XML. This system uses KUKAVARPROXY, a server developed in Visual Basic to implement the Kuka CrossComm interface and allow for interaction with real-time robot control processes. JOpenShowVar, which is written in Java, runs as a client on a remote computer connected to the Kuka controller via TCP/IP. The Java-based platform allows for high-level programming and use of third-party libraries. However, as discussed above, when accessed via TCP/IP through the KUKAVARPROXY server, it creates unavoidable delays in real-time access to the robot's data; thus, it can only be used for soft real-time applications. In addition, as this system is based on Java, it makes if difficult to work with C#.NET components in Grasshopper and Rhino .NET Software Developers Kit, the target application for the creative industry.

Another interface currently available for Kuka robots is mxAutomation, which allows for real-time communication with Kuka industrial robots (Munz, Braumann, and Brell-Cokcan 2016, Braumann and Brell-Cokcan 2015). The mxAutomation software package was created in collaboration with Siemens; it has two main parts, a server program that runs on the Kuka robot controller (KRC) and a robot control program with a client library that runs on a remote computer. These two parts are connected via either field buses such as EtherCAT or a user datagram protocol (UDP) and Ethernet. The authors developed a custom client software that connects the mxAutomation library with Robots in Architecture's KUKA|prc framework that runs in the Rhino/Grasshopper environment. This allows for the exchange of data between a remote computer with KUKA|prc and a robot. While the system offers a high-level programming interface for the user with promising applications, it requires the use of the mxAutomation software package.

## 3.4 Method

This research proposes a framework for human-robot interaction that has two main elements: 1) an adaptive robot control system based on sensor feedback and 2) a design-fabrication library. The main advantage of our proposed framework compared to other research efforts to develop adaptive robot control is the use of real-time feedback from a scanning system, as well as the reading/writing of data to/from a design-fabrication library.

The present work discusses the first element of the framework, the adaptive robot control system. The system architecture diagram in Figure 3-3 offers a high-level view of our creation and its major components. The adaptive control system uses the following hardware and software elements as a testbed for the framework. For hardware, we employed Industrial Kuka robotic arms with either KRC2 or KRC4 operating systems and Microsoft Kinect as the 3D scanner. For software, we used Rhino 3D, Grasshopper, the KUKA|prc robot programming plug-in for Grasshopper, Kuka RSI, and Kinect Fusion library. The only prerequisite is that the user must have a solid understanding of computer networking and the KRL programming language.

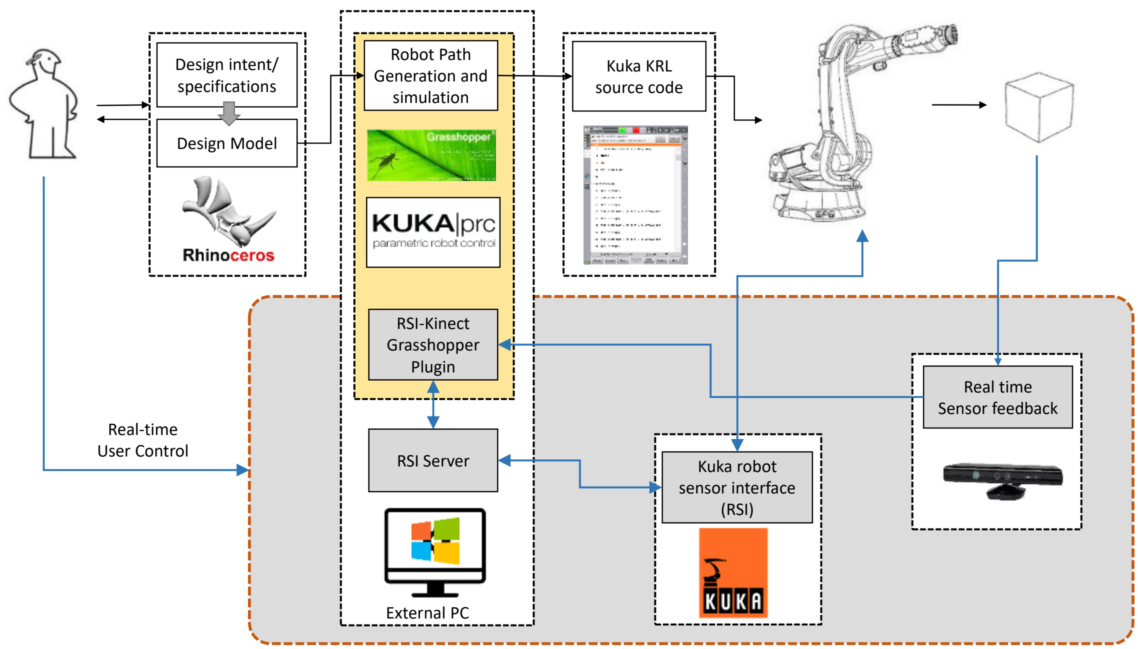


**Figure 3-3. Proposed real-time control system architecture for Kuka industrial robots. The server runs as a client on a remote computer. The server interacts locally with Rhino/Grasshopper and remotely communicates with Kuka KRC via TCI/IP. Feedback data from the Kinect 3D scanner is received by Grasshopper.**

The design of our system is based on the following choices:

- Target users: This system is intended for use in design and fabrication processes, and thus architects and designers are the target users. As a result, it is important that the system sync with and run on CAD applications such as Rhino/Grasshopper and Autodesk Dynamo for visual programming purposes.
- Speed: Although Microsoft Windows computers do not provide hard real-time communication, it was desirable to minimize the communication lag between the robot control system and remote computer as much as possible. Industrial robots have real-time constraints, and hence maintaining the speed of the application is imperative.
- Native packages: This system employs Kuka's own software package, Kuka Real-time Sensor Interface (RSI), for real-time communication and to ensure system compatibility.
- Flexibility: Our system provides a structure and system architecture that offers the future possibility of including third party libraries.

Our overall architecture consists of three high-level modules: an RSI-Grasshopper module, Kinect-Grasshopper module, and KUKA RSI server (see Figure 3-4). We outline the design of each module below.

### *3.4.1 KUKA RSI Server*

KUKA robots allow for real-time control via the KUKA.RobotSensorInterface from an external PC over an Ethernet connection. To enable this, the user must write a UDP-based network server on an external PC in the programming language of their choice and provide the Internet Protocol address of the server to the robot via the RSI configuration XML. This allows for bidirectional communication between the robot and server, and thus for robot motion to be corrected via XML-based instructions. We developed our RSI server in the Python programming language due to its ease of use in experimentation and abundance of libraries of network operations. Our implementation supports both KUKA KRC2 and KRC4 robots.

Our RSI server spawns three sub-servers: robot, read, and write server. These are essentially sub-processes that communicate among one another. The robot server connects to the KUKA robot over UDP and always responds to the robot within the 12-millisecond time limit, in order to maintain a hard real-time constraint and keep the connection active. This also allows the read and write servers to perform long-running operations independently and not violate the response time constraint. The robot server checks for any new input at each cycle before transferring the input or a standard response without correction to the robot, while also always updating the new robot configuration via its internal data structures. The read server reads the RSI data from the robot server's internal structures and provides it to the user in a JSON format for display or logging. The write server accepts input from the user in the form of a JSON of per-axis corrections and transforms this input into a JSON format that is then sent to the robot server; there, it is used to encode valid XML and then is sent to the robot. We use JSON to communicate between the three sub-servers due to its ease of use with Python and many other high-level languages (such as Matlab or C#) and relatively low memory requirements (as compared to XML). All communication among the three sub-servers is done using inter-process messaging queues.

To ensure the safety of the robot and not violate its torque correction limits while performing robot corrections via the RSI, we chunk all corrections into smaller corrections of 2 millimeters or less, and generate the appropriate number of UDP packets; these are sent in batches. This also provides smoother path corrections for the robot and allows for better feedback to the user. The chunk size is a configurable variable in our program, permitting either very slow and small or large and rapid motions, as desired.

From the KRL programming end, the RSI object is created and correction limit set to the approximate boundaries of the workspace in which the robot will operate. This ensures that the KRL program does not give an error due to limit violations. The RSI object is enabled when the program is run, allowing it to communicate with the RSI server.

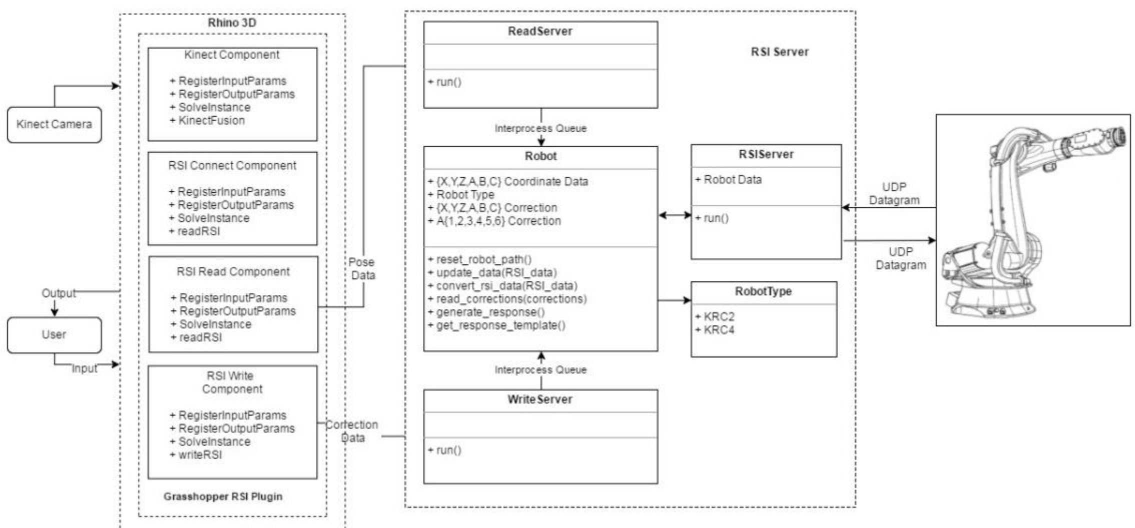


**Figure 3-4. Kuka RSI server and sensor feedback system structure.**

### *3.4.2 RSI-Grasshopper Module*

To allow for Rhino 3D to transfer user updates to connected robots, we developed an RSI-Grasshopper module using the Grasshopper plug-in (see Figure 3-5). This module allows for the integration of Rhino 3D with the RSI server by linking Rhino 3D UI elements with data in the RSI server via tool center point (TCP) Ethernet connections. The read and write sub-servers in the RSI server provide an external interface for our RSI server, allowing us to send the robot corrections and receive the RSI data. In this module, we create the TCP requests to transfer data to the write server, read the robot pose from the read server, and update the Rhino 3D user interface, thus allowing the user real-time control and updates. The server and added module in Grasshopper can together receive and see the robot's actual position and send toolpath corrections in real time.

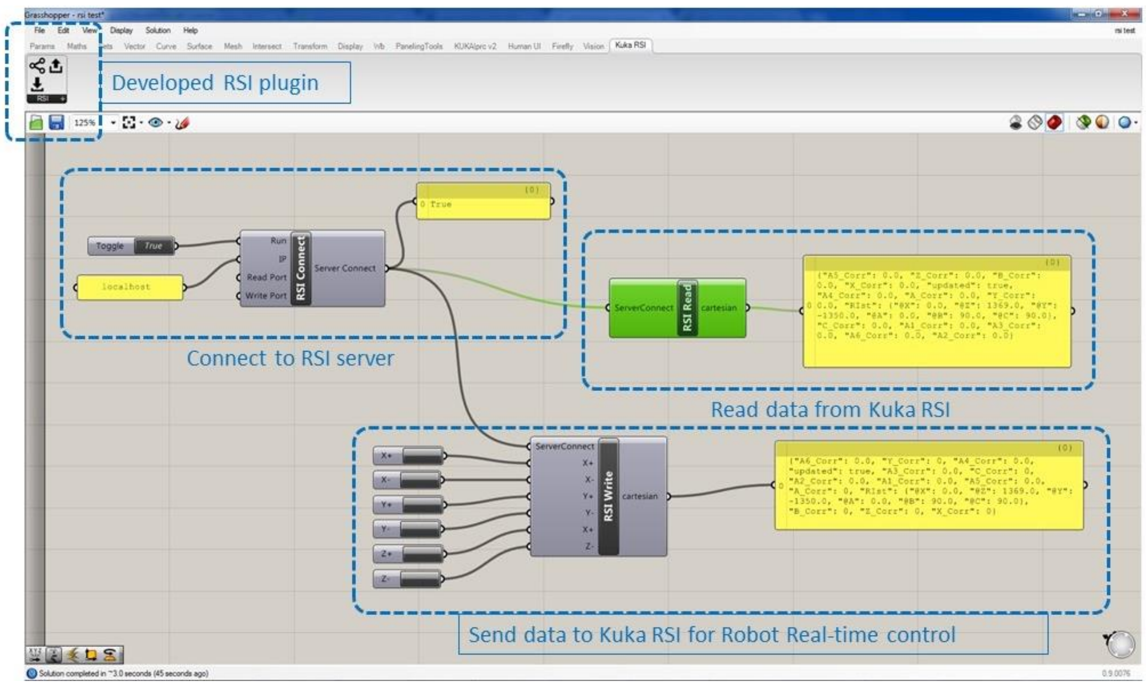


**Figure 3-5. Grasshopper plug-in developed for the present research: server connection, data read, and data write.**

### *3.4.3 Kinect-Grasshopper Module*

To allow for visual feedback of the object being fabricated, we integrated Kinect sensor feedback into Rhino 3D by developing a Kinect-Grasshopper plug-in module (see

Figure 3-6). The Kinect acts as a 3D scanner to generate a real-time point cloud of the object as feedback to the user. Using the Kinect Fusion Library in the C# programming language, we developed a plug-in for Grasshopper that brings the Fusion depth stream to Grasshopper, providing a 3D point cloud of the actual physical environment of the part under fabrication inside the Grasshopper environment. Using this 3D view, the user is able to move the robot TCP to an actual point in physical space via the RSI-Grasshopper module and use the visual feedback from the Kinect window to guide the robot.

As future steps in service of completing this system, we intend to develop a structure where by overlapping the mesh generated from the Kinect point cloud data onto the existing CAD model, it will be possible to measure deviations between the expected 3D and actual physical forms.

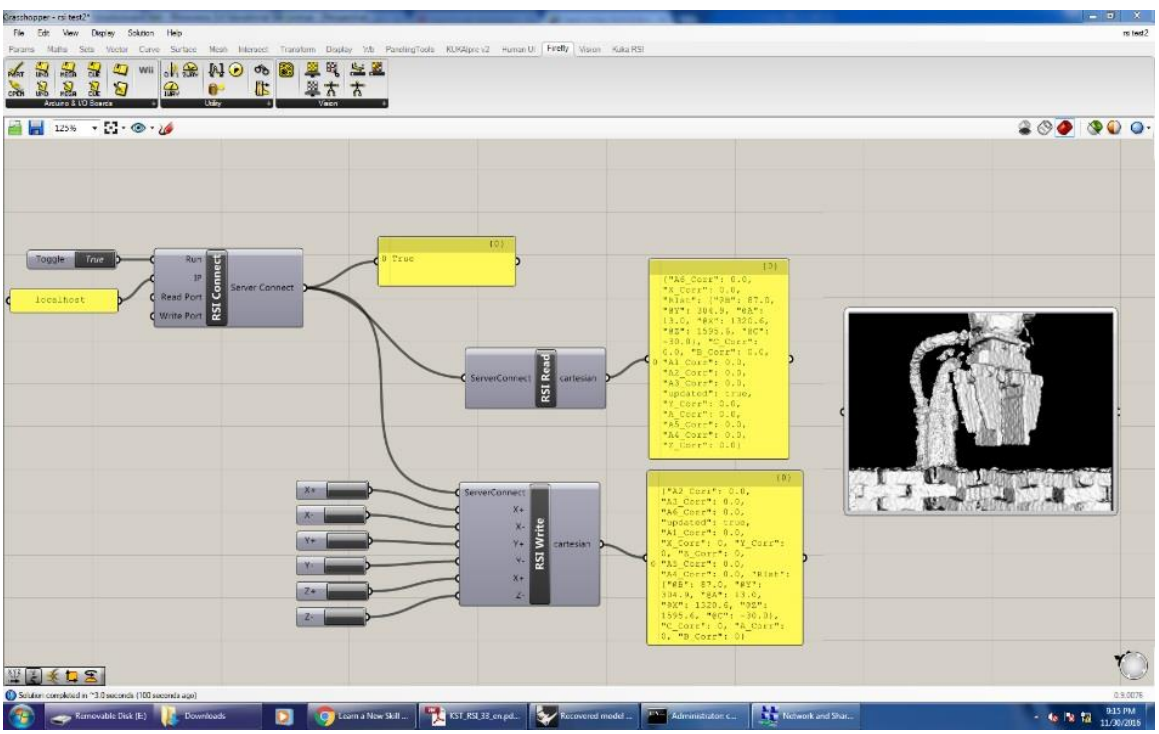


**Figure 3-6. Grasshopper and Kinect Fusion.**

## 3.5 Conclusions and Future Steps

In this research, we described the first part of a real-time framework for robotic fabrication, particularly the integration of a Rhino/Grasshopper CAD modeling and visual programming environment with an industrial robot, allowing for instant feedback during the fabrication process. We described the architecture of our system, which provides a means of maintaining the real-time constraints required by such a robot and provides the end user with an efficient means of completing their task without damaging the robot or its surroundings. Visual feedback in terms of both robot and 3D depth data from a Microsoft Kinect with a Kinect Fusion library is used to provide comprehensive information throughout the process.

In the next step in the development of this comprehensive human-robot interaction framework, we will focus on producing a design-fabrication library that will utilize this adaptive robot control system to store various design and fabrication information in its database. This library will assist designers In the conceptual design stage with decisionmaking regarding the adequacy of their design choices, detailing, materials selected for robotic production techniques, and end effectors. This design library will be instantiated incrementally based on experimentation and human expert knowledge. It will grow over time to incorporate data from ongoing experimentation by multiple users enabled by the system described here. In addition, a user study on the human-robot Interaction aspects of the system will be performed. This study will compare scenarios of identical fabrication processes with and without the assistance of the adaptive robot control system, measuring aspects such as time, success rate, number of successful attempts, and quality of the final product.

# CHAPTER 4. ROBOTIC FABRICATION LIBRARY

## 4.1 Introduction

Creative design and fabrication in architecture is an abductive and highly exploratory process. Architects investigate and solve complexities related to new forms, materials, and fabrication mechanisms and unpredictable construction environments by relying on a reciprocal investigation of conceptual digital and material forms. However, working with robotic technologies imposes a divide between the acts of designing and making. The current process of design for robotic fabrication is mostly a unidirectional workflow in which the designer must predict the material state, tool selection, fixture positioning, and robot motion, usually based on prior experience. Human-robot collaboration is the foundation of efforts to enhance robotic applications in creative architectural fabrication processes. In a successful collaboration, human designers are the creative force in this process. Their vision is executed by machine fabricators via unique specifications in order to produce the intended artifacts.

This dissertation investigates how feedback on the state of materialized and embodied artifacts can be incorporated into various solutions for enhancing human-robot collaboration workflow. Feedback on a material's state can be captured using digital tools such as sensors, scanners, and cameras, or by relying on human analog sensing such as vision and touch. Feedback is required to create a complete closed-loop interactive system for bidirectional design-to-fabrication workflow (Garcia del Castillo Lopez 2019, Munz, Braumann, and Brell-Cokcan 2016, Brugnaro and Hanna 2017, Stumm et al. 2016).

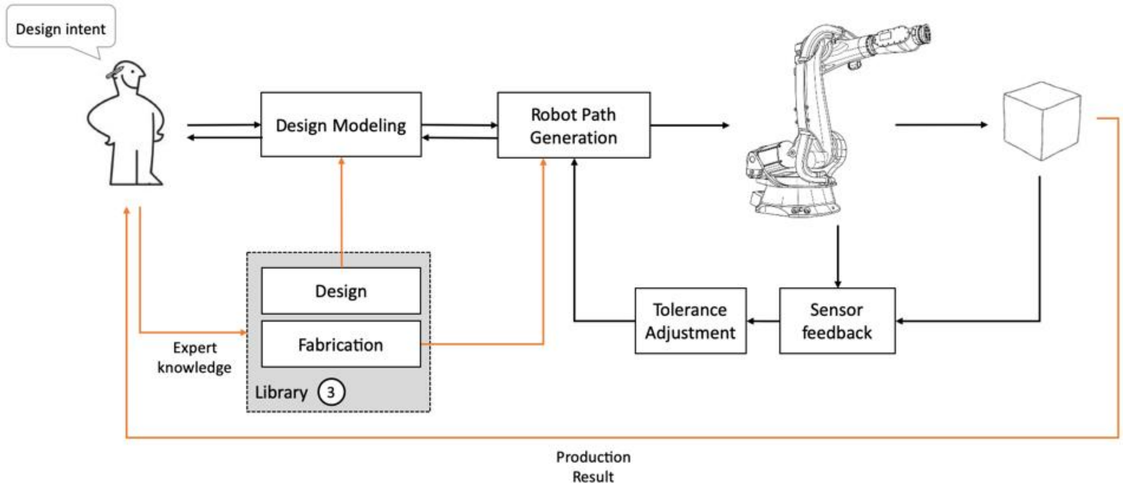


**Figure 4-1. Design and fabrication library for human-robot collaboration.**

Captured feedback can be used in two major ways to facilitate collaborative design-to-fabrication processes (see Figure 4-1). First, feedback enables real-time adaptive control of robots by gathering data from various sensors and feeding that information into the system, modifying the robot's operation based on the material's conditions. Adaptive control systems help fabricators program processes that integrate uncertainty as a part of fabrication by capturing and reflecting feedback from both human and robot collaborators. Various studies have reported on a number of adaptive control systems developed for creative fabrication (Garcia del Castillo Lopez 2019, Brugnaro, Figliola, and Dubor 2019, Sharif, Agrawal, and Sweet 2017). Second, the feedback captured and recorded from digital sensors or human inspection and experience can be formalized and stored in a library for future reuse. A well-defined and broadly adopted design and fabrication library would assist users with decisionmaking by offering solutions for processes, materials, geometries, tools, and fabrication instructions.

As compared to adaptive control, there are few research precedents that have investigated a library or database component for human-robot collaborative systems. Each year, numerous studies on robotic fabrication in architecture are completed. Work is evolving from both controlled laboratory experiments and onsite robotic installations (Lloret-Fritschi et al. 2018, Schwinn 2017, Peters and Belden 2014). While these projects propose creative solutions for integrating sensor input and real-time feedback into the design and fabrication process, the solutions tend mostly to be project-based. Moreover, there is no standard data structure format for information capture nor ability for others to reuse project-based knowledge in future endeavors. Much of the information on creative design solutions and fabrication methods generated during these projects is not recorded. Papers and reports tend to describe overall strategies and levels of success, but few include implementation details. The lack of a centralized and accessible knowledge repository results in the loss of valuable design knowledge that could otherwise be reused and built upon.

There are a few notable projects that are helping define the requirements for design and fabrication databases. Dritsas discussed the concept of a library in the form of a plugin for Grasshopper in order to help designers discover ideas for design and fabrication (Dritsas 2015). However, the study does not provide much detail on the implementation or application of such a system. Maidin et al. proposed the use of a design feature database as a knowledge-based tool to guide designers at the creative conceptual design stage for additive manufacturing (AM) processes. While this database considers various materials and machine options for AM, its main focus is on the buildability of various design options (Maidin, Campbell, and Pei 2012).

In this chapter, we propose a high-level structure for a library data model targeting robotic fabrication. The library is defined as a structured knowledge database of successful fabrication methods and their requirements. The main goal of the library is to provide users with process data to help them in their decisionmaking. Related data from prior projects with similar fabrication processes, tools, setups, materials, and instructions would help users avoid errors, shorten the time required to produce useful parts, and develop new ideas by learning from past experiences.

Like any other computer application, a digital fabrication library can only store and work with explicit knowledge that can be represented as words and numbers and communicated systematically through data, scientific formulae, specifications, and manuals (Eraut 2000, Nonaka and Konno 2005). However, designers and fabricators' expert knowledge and personal skills are mostly tacit and difficult to express, formalize, and share with others (Reber 1989, Bernal 2016b). Tacit knowledge is personal and often gained through observation, induction, and participation (rather than formal inquiry). While a maker's knowledge in working with a material is tacit in nature, this knowledge has to be translated into an explicit structure so that it can be stored in a digitized database and used to operate robots and other computer-controlled tools.

Consequently, the proposed structure for this database is a roadmap according to which information can be captured and connected. The remainder of this chapter discusses the main elements of the library (i.e., the entities) and the relationships among them. It is anticipated that the proposed data model will facilitate a stepwise modeling of the entire fabrication process such that the user’s tacit knowledge can successfully be stored.

## 4.2 A Process-centric Data Model

The core concept supporting the design and fabrication library is the fabrication process, which refers to the act of making. Thus, this research adopts a process-centric approach as a systematic method for organizing data for use in this type of library. This dissertation builds upon the previous work on process modeling by researchers at the Digital Building Lab at the Georgia Institute of Technology (Eastman et al. 2009, Barak et al. 2009, Eastman, Lee, and Sacks 2002). Bernal proposed the use of model-based engineering concepts for capturing, structuring, and representing the tacit knowledge embedded in design patterns so that this knowledge could be reused for the generation of design alternatives (Bernal 2016a, b). Valdes et al. demonstrated a diagrammatic language that helped students represent and reflect on their intended fabrication and assembly activities (Valdes, Cavieres, and Gentry 2013). This method borrowed from the syntactical structure of conceptual maps (CMaps) developed by researchers at the Florida Institute for Human and Machine Cognition (IHMC). Concept maps are graphical tools that help users organize and represent relationships among concepts to facilitate easy navigation and sharing (Novak and Cañas 2006, 2007).

The entity-relationship (ER) model as been adopted as the diagrammatic language for the conceptual modeling of this fabrication library (Navathe and Elmasri 2010). In the first step, the five main entities of the library are introduced: Process, Material, Geometry, Tool, and Instruction (see Figure 4-2). This step is followed by a detailed definition of the relationships among the entities (see Figure 4-3). An example from a robotic sheet metal folding process is provided to illustrate the concepts discussed (see Figure 4-4 to Figure 7-12).

However, it should be noted that the attributes required for detailed definitions of the entities are not represented at this stage. Identification of these attributes is an endeavor that requires further study of various processes, tools, and material types. Consequently, several chapters of this dissertation (Chapters 5 to 7) focus on various case studies of different fabrication methods, tools, materials, and instructions in order to provide more in-depth insight into these requirements. The goal of this study is to illuminate the level of detail required for defining a comprehensive library structure.

### 4.3 Structure of the Library

This section offers a high-level articulation of the library structure. There are five major entities in the library. Each of these components is further broken down into sub-types. The proper connection and synchronization of these entities defines the essential structure for correctly representing fabrication knowledge.

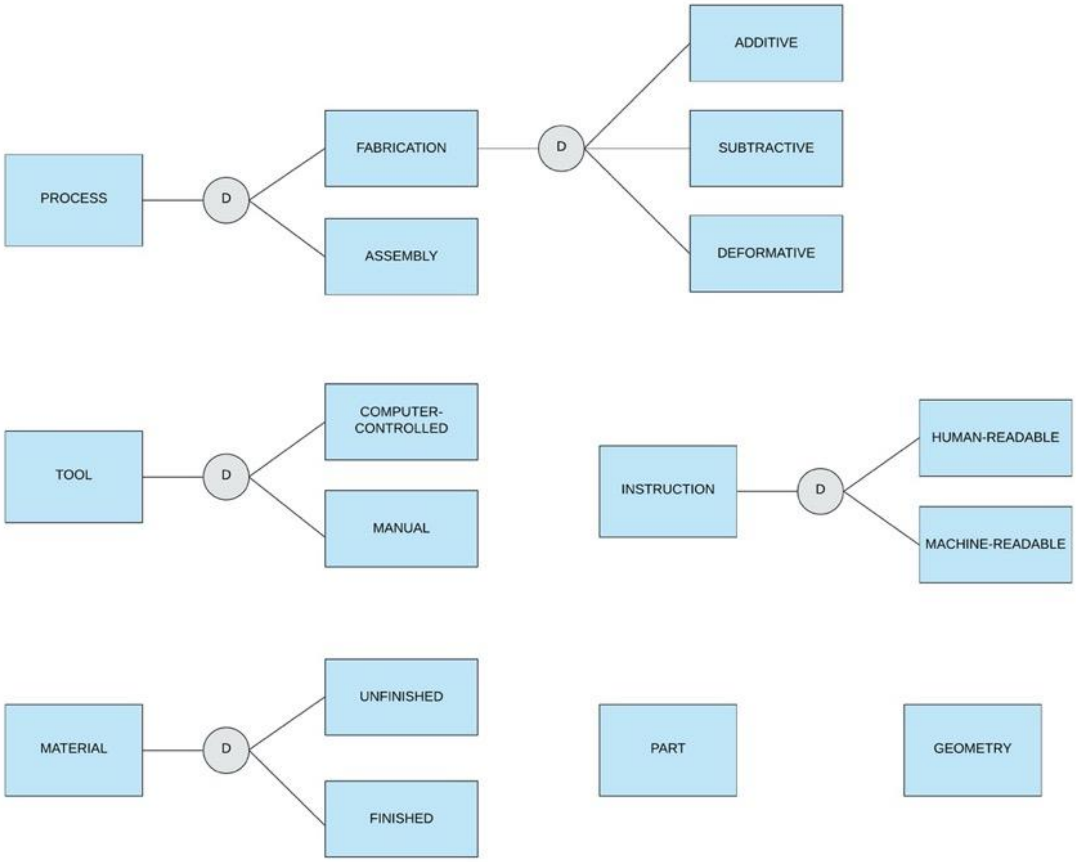


**Figure 4-2. EER diagram of the library's entities and subtypes.**

The **process** entity is the core of the library. It represents any physical activity that is required to construct the final form of a component from raw materials, based on a set of specifications and using various tools. Processes can be categorized into two primary sets of activities: fabrication and assembly. Fabrication processes are additive, subtractive, or deformative operations that transform raw materials and parts into intended components. Assembly processes are required for connecting discrete fabricated components by using connector parts and materials (i.e., screws, bolts, adhesives, locking joints) to form the final product (see Figure 4-2). The process itself describes activities at a high level, such as in metal bending, bricklaying, and woodworking. It is necessary to break each process down into multiple process steps in order to accurately capture the entire activity. Process steps are granular activities during which the part changes in form and is moved closer to the desired outcome. For example, a metal bending process can be itemized into: pick-up, move, place, bend, and rotate process steps (see Figure 7-12). A woodworking process can include process steps such as mill, drill, cut, and carve. In a complete fabrication or assembly process, the process steps are repeated multiple times in various orders. Each can have different instructions for operation. A robotic metal folding process can have several continuous sequences of a series of process step operations (e.g., pick-up, rotate, place, bend) to represent a complete deformative fabrication process. Various robotic fabrication processes and process steps are discussed in detail in Chapters 6 and 7.

The **tool** entity is the next component of the library structure. Tools can be either computer- or human-controlled (see Figure 2). Examples of computer-controlled tools include robotic arms, CNC mills, water-jet cutters, and laser cutters. Human-controlled tools can either be simple implements such as chisels, hammers, and screwdrivers, or electrical devices like table saws and chop saws. All process steps are performed by one or multiple synchronized tools. These tools are operated based on a set of instructions with the goal of producing a desired geometrical form. Each tool has various affordances and limitations that should be considered during fabrication and assembly activities. For example, the tools for a robotic straight-line sheet metal folding include one industrial robotic arm enhanced with a gripper end-effector and one additional stationery gripper (see Figure 7-8). Further explanation of the various tools of fabrication is offered in Chapters 6 and 7.

The **geometry** entity is defined in close relation to tools and instructions. In any process step, tools perform fabrication activities to affect a materialized change in the parts in order to satisfy the geometric data that the designer has defined. Every tool can operate with either a 2D or 3D geometry. The five sets of 3D geometries that are required for completing the robotic sheet metal folding of a part are illustrated in Figure 7-11 and discussed in detail in Chapter 7.

The **material** entity represents the physical constituent required for a process. Both unfinished and finished materials are used in fabrication and assembly activities (see Figure 4-2). Examples of unfinished materials include sheet metal, timber wood, sheet plastic, and uncured concrete. Instances of finished materials include made-to-stock components such as screws, bolts, and nuts. Tools are used to transform unfinished material into desired geometric forms. Finished materials are mostly used as fasteners for assembly. A detailed analysis of the attributes and proposed data structure for a sample construction material (i.e., masonry) can be found in Chapter 5.

The **part** entity depicts both the input and final output of a process step (see Figure 4-2). The input for the first process step is usually an unfinished material such as a piece of timber wood or sheet metal. Cutting the process step would convert it into an output part with specific geometric definitions. The cut timber wood or sheet metal part becomes an input part in the second process step, which can be either drilling for wood or bending for sheet metal.  Figure 7-9 shows the input and final output parts for the robotic sheet metal folding example (see Chapter 7 for details).

The **instruction** entity goes hand-in-hand with tools and geometries. Instructions can be either machine-readable (e.g., KUKA KRL or G-Code) or human-readable (e.g., shop drawings) (see Figure 4-2). Their primary function is to assist the tool operator (either computer or machine) in performing a task. Instructions provide guidance adequate to transform a part into the desired geometric shape in a fabrication process or connect multiple parts in an assembly activity. Figure 4-9 presents an example of the instructions that go with every process step for a full set of sheet metal folding operations. Chapters 6 and 7 provide a detailed explanation of the instructions for two case studies: a robotic masonry wall fabrication and robotic sheet metal folding.

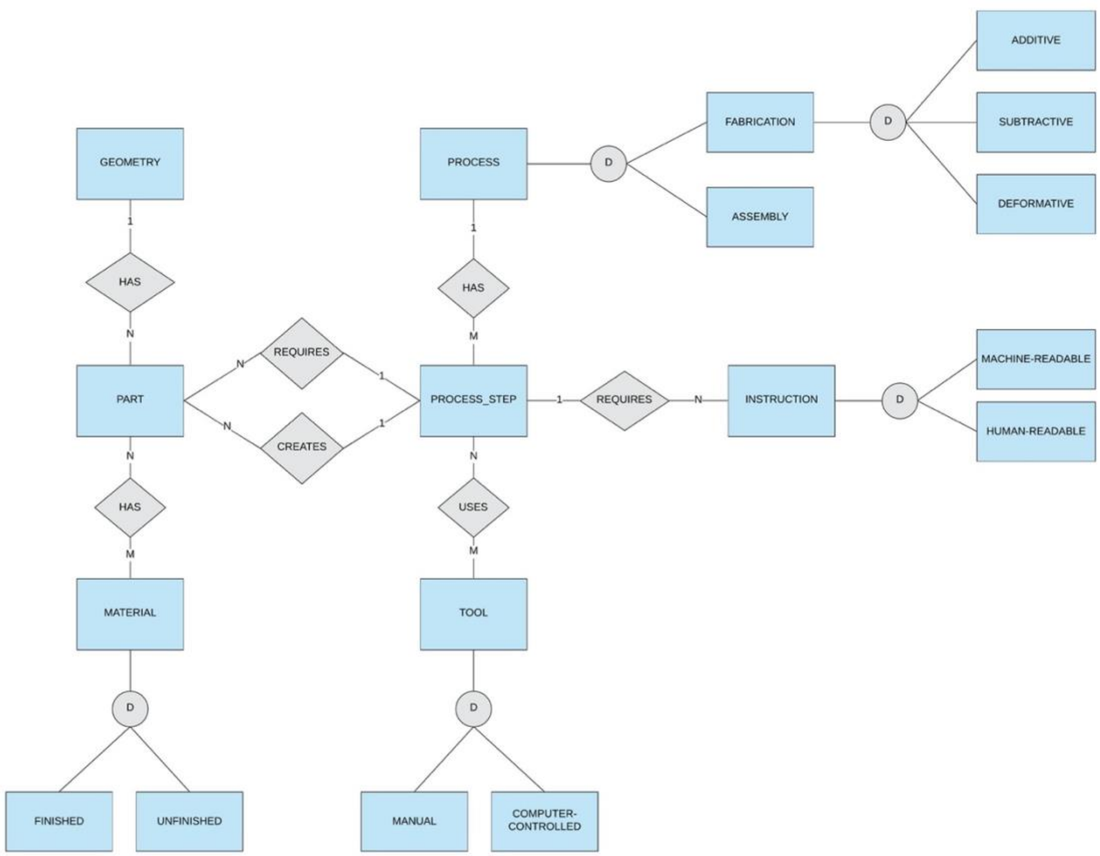


**Figure 4-3. ERR diagram of the process-centric library.**

All of these major sets of entities are connected to one another, together forming a connected process model. The diagram in Figure 4-3 illustrates the high-level structure of the library. The proposed data model facilitates stepwise modeling of the entire design and fabrication process. In this system, the model is structured around the fabrication process as the core element of the library. It is supported by other factors that are also required for its realization. Every process is composed of several process steps. Tools perform the process task by following a set of instructions. Tools modify the input part to produce a final output part that matches the intended geometry. Figure 4-4 presents an example of a robotic sheet metal folding process (P01) that is comprised of four bending process steps (PS01 to PS04). The input part (see P01 in Figure 4-4 and left image in Figure 7-9) goes through four steps of deformation (P02 to P04), till it finally matches the final intended geometry (see P05 in Figure 4-4 and right image in Figure 7-9). The same set of tools, a robotic arm with a gripper end-effector, and stationary gripper are used for all four process steps (see Figure 4-4 and Figure 4-8). A set of instructions for moving the robotic arm and closing and opening the grippers is required to accompany every process step (see Figure 4-9).

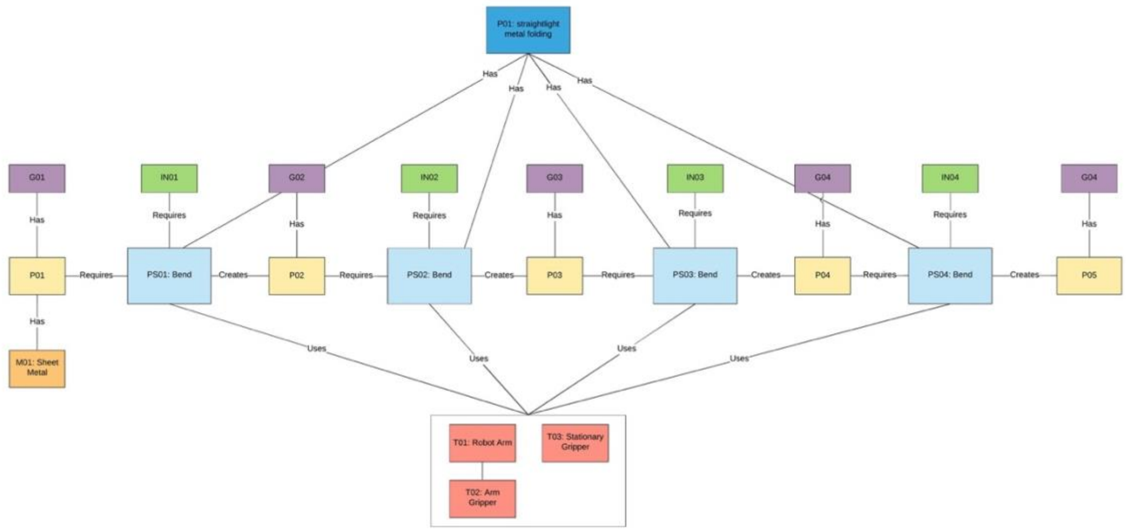


**Figure 4-4. Expanded high-level conceptual diagram for the robotic sheet metal folding process.**

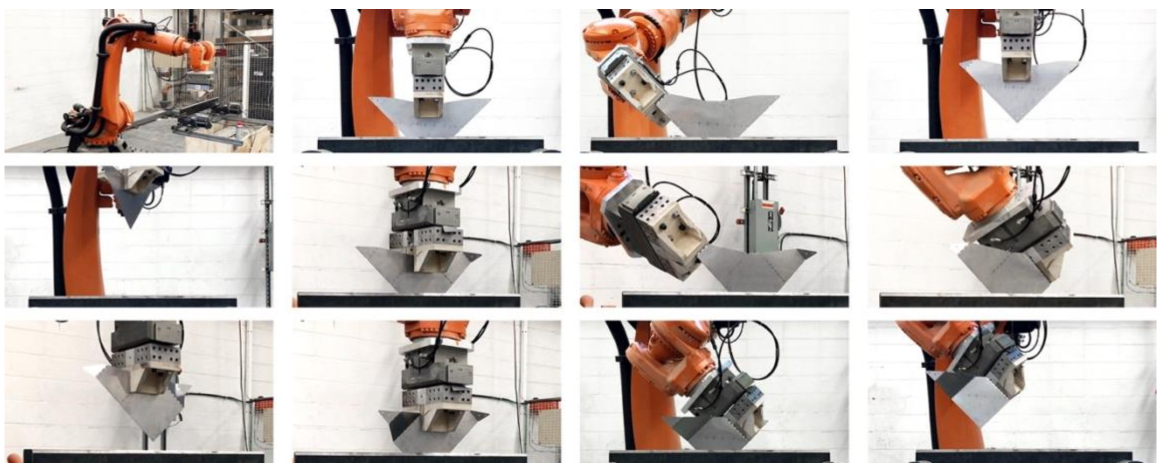

**Figure 4-5. Stepwise sequence of steps in the robotic sheet metal folding process (PS01 to PS04).**

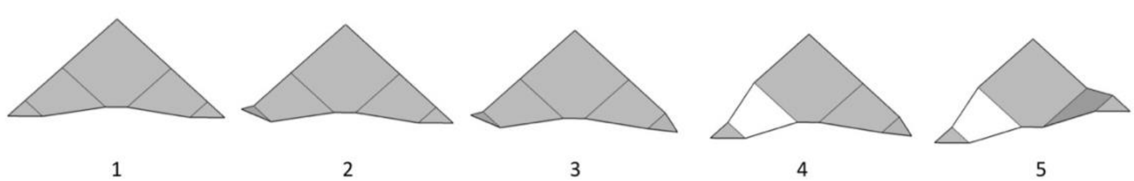


**Figure 4-6. Stepwise geometries for the robotic sheet metal folding sequence (left to right: G01 to G05).**

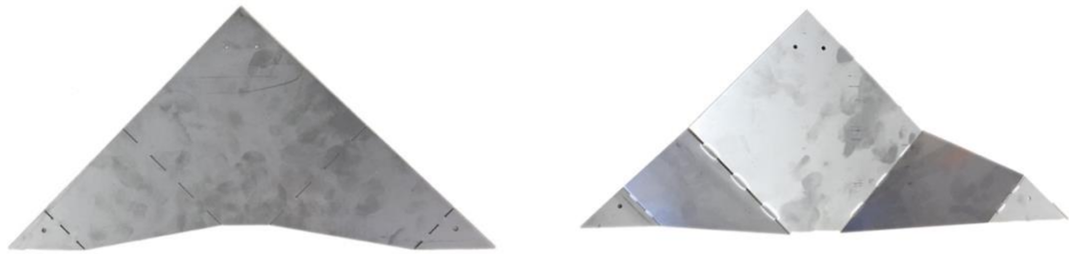

**Figure 4-7. Flat input part (P01) before robot operations (left) and folded output part (P05) after the robot folding process (right), producing the desired final geometry.**

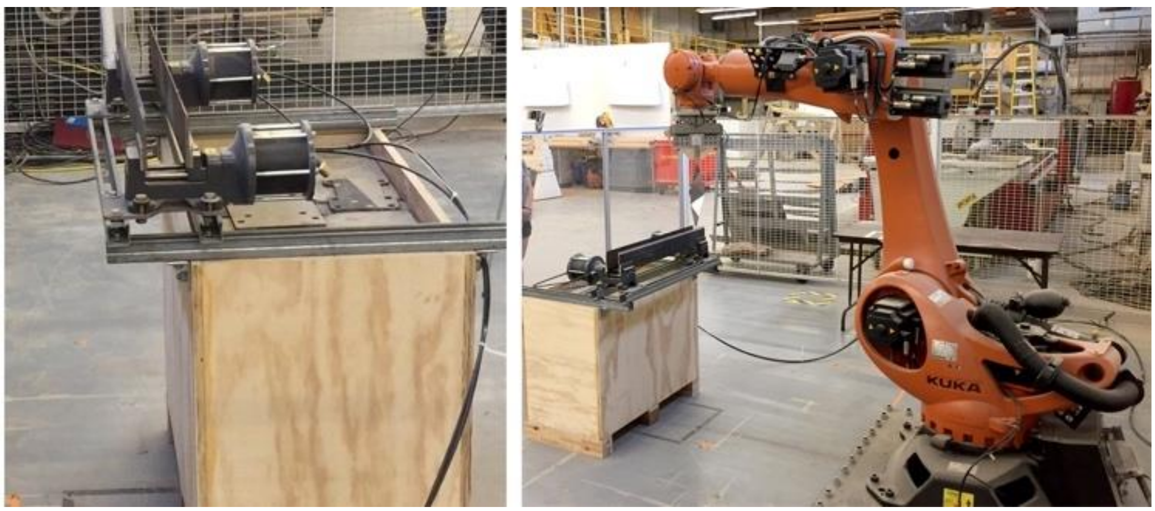

**Figure 4-8. Required tools and environment setup for the robotic sheet metal folding process (T01: robot arm, T02: arm gripper, and T03: stationary gripper).**

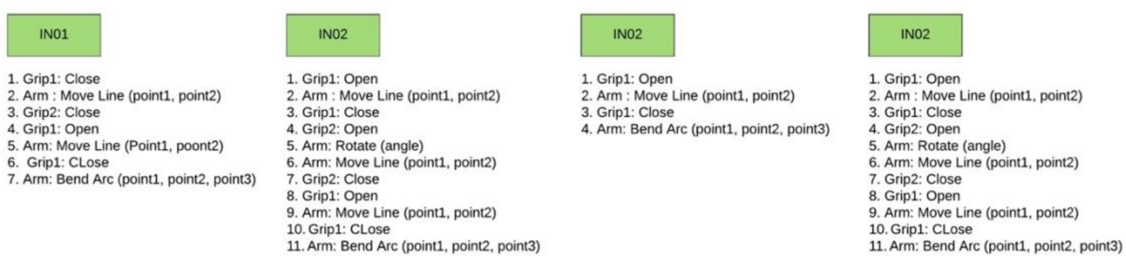


**Figure 4-9. Stepwise high-level instructions for robotic sheet metal folding.**

## 4.4 Conclusion

This chapter emphasizes the role of captured and reused feedback in developing a human-robot collaboration framework for enhancing bi-directional design and fabrication processes. In addition to using real-time feedback during the act of fabrication, feedback can be captured, stored, and reused to preserve knowledge in a library. A well-defined and broadly adopted design and fabrication library can assist users with decisionmaking by offering fabrication process alternatives, as well as material selection, detailing, and machine setting options. Consequently, a high-level process-centric data model for such a library is proposed. This library and data model are intended for architectural design-to-fabrication processes focused on either creative form development or the fabrication of customized parts and prototypes in small batches. It should be noted that this model of fabrication has inherently different characteristics and needs from those of industrial production requirements.

While a well-defined structure for this type of library is needed, the structure alone is not sufficient. In addition to an adequate data model and proper technological implementation, the success of a database relies on the social aspect of its adoption and user contributions. The data required for a functionally successful library should be gathered from experimentation and human expert knowledge inputs that are added to the database incrementally over time. This type of library requires a proper data model and database system, but those alone will not make it successful without a community of users to add information and draw upon it.

# CHAPTER 5. PARAMETRIC PRODUCT LIBRARIES: DEVELOPMENT AND IMPLEMENTATION OF A MASONRY UNIT DATABASE


## Abstract

This research describes the development of an industry standard data model for concrete and clay masonry units based on the requirements defined by the Building Information Modeling for Masonry Initiative (BIM-M). We begin with a review of classification systems and building object model libraries, examining examples of the product standardization that has taken place as a result of this classification. Data requirements are explained from a stakeholder perspective, beginning with early-stage design where masonry units are selected based on initial design criteria, and moving to installation where contractors use unit data to procure, estimate, and install units. The data model represents a broad range of information beyond simple geometry, in order to facilitate design and construction activities specific to masonry. In this work, we describe a process for the parametric representation and storage of digital masonry units that allows for speedy retrieval of such units from a database. A method for storing and mapping colors and textures onto the units for product visualization is also presented. Finally, we discuss the commercial implementation of the data model proposed here, and demonstrate links between the database and both web and BIM platforms. Using masonry as an example, we show how complex building objects that have heretofore been outside the scope of BIM models and processes can be integrated into modern design and construction procedures.

## 5.1 Introduction

Building Information Modeling (BIM) supports industry standard data models that facilitate information sharing and exchange among architecture, engineering, and construction (AEC) stakeholders. In collaboration with manufacturers and providers of building products and systems, the programmatic requirements of building design and construction can be captured and used in the development of data models (Eastman et al. 2011). These data models encapsulate and codify industry standard product descriptions and enable queries across BIM software platforms. In the AEC industry, these data models and related software and business processes are core BIM elements.

Different sectors in the construction industry, including structural steel AISC (Crowley and Watson 1997), precast concrete (PCI) (Eastman, Lee, and Sacks 2003), and cast-in-place concrete ACI (Barak et al. 2009) have recognized the need to implement BIM standards and adopt the required processes for development of data models, thus allowing for the accurate exchange of construction information among designers, engineers, contractors, and fabricators. Among these materials-specific data models, structural steel is the most mature; the standardization of steel shapes in the late 1800's formed the basis for the steel components used in buildings today (Standard Specifications for Structural Steel 1896).

### *5.1.1 BIM Standards and Processes in the Masonry Industry*

Different from the structural steel and precast concrete industries, the adoption of BIM by the masonry trade has been hindered due to the complexity of masonry products and construction systems, as well as by a lack of modeling schema for masonry units, walls, and buildings. Early work by our research team demonstrated the potential for linking parametric models of concrete masonry units (CMU) directly to structural analysis (Cavieres, Gentry, and Al-Haddad 2008). Monteiro et al. discussed the possibility of modeling the architectural aspects of masonry in AutoDesk Revit and identified a lack of data models for masonry components and assemblies (2009). Knight and Sass developed parametric modeling techniques for custom block systems, along with a grammar to describe their order of assembly (2010). Nawari established an initial concept for representing masonry walls in IFCs and identified the basic data requirements for the structural analysis of masonry using that form of representation (2011). More recent work by our team introduced the concept of an object-oriented representation for masonry walls (Cavieres, Gentry, and Al-Haddad 2011).

On the construction side, stakeholders have questioned the value and applicability of BIM applications and processes to masonry-specific tasks. The 2014 McGraw Hill SmartMarket Report showed that masonry and concrete contractors were among the least likely to adopt BIM in their practice (Jones and Bernstein 2012). A common response to questions about BIM has been that the “functionality does not apply well enough to what we do.” In 2015, a group of mason contractors surveyed by our colleagues indicated that they would adopt a 3D modeling tool that addressed site logistics (with an 81.5% positive response) and project scheduling (with an 88.9% positive response). In the same survey, however, 56% of mason contractors stated that they had not yet used BIM. When queried regarding their impressions of BIM, the most common response was “BIM does not apply to us” (23%), followed by “BIM is a good tool but with qualifications” (17%) (Kinateder 2015).

In light of this situation, the Building Information Modeling for Masonry Initiative (BIM-M) was organized in North America in 2013 in effort to advance the use of BIM for masonry. Stakeholders supporting the initiative include masonry-related professional societies, materials manufacturers, contractors, and labor unions (http://www.bimformasonry.org). In its first year, BIM-M published a roadmap for establishing the requirements for masonry data models to support the design, planning, procurement, and construction of masonry buildings (Gentry, Eastman, and Biggs 2013). The roadmap identified three key projects for the initial focus of BIM-M: (1) development of a standard representation for masonry units (the activity reported in the present research), (2) development of a standard representation for masonry walls and systems (Cavieres and Gentry 2015), and (3) the completion of case studies to establish "current state" and "future state" practices for the application of BIM in masonry projects (Lee et al. 2015). This study focuses on the development of a masonry unit database (MUD), drawing from the stakeholder analyses and wall schema development in related projects.

### *5.1.2 Masonry Unit Database*

The design and construction of masonry buildings involves the exchange of substantial technical information among a wide range of project participants throughout the lifecycle of a building project; such participants include architectural designers, structural and energy analysts, masonry manufacturers and vendors, masons, and mason contractors. A primary goal of the MUD project was to facilitate this process by developing an infrastructure for allowing the electronic exchange of technical information between various BIM software applications. Effective data exchange required the development of a data structure to represent masonry buildings, with intelligent masonry objects enhanced with embedded information and logic regarding the masonry objects themselves and their relationships to other objects in the building model.

MUD is a type of building product library specifically tailored for masonry products such as clay brick, concrete masonry units, cast stone, and cut stone. MUD can be compared to the database of hot rolled steel shapes created by the American Institute of Steel Construction (AISC) (AISC 2017) and British Standards Institute (BSI), which forms the data foundation of structural steel modeling and fabrication software (Crowley and Watson 1997). In any BIM masonry building project, MUD can serve as an integral part of the project model's development, as it provides access to the details of various masonry products such as unit geometry, material properties, color, and texture.

### *5.1.3 Building Product Libraries*

Building product information is required by different project actors throughout the lifecycle of a building project, from design to the construction and eventual maintenance stages (Eastman et al. 2011, Afsari and Eastman 2014, Costa and Madrazo 2015). In any building project, building products account for 40% to 70% of the cost of construction (Jaśkowski, Sobotka, and Czarnigowska 2018, Lu et al. 2018). There are thousands of building product manufacturers, and each produces a variety of items to fulfill a wide range of architectural and engineering requirements. Traditionally, information describing these products has been delivered in printed catalogs. More recently, the same type of material has become accesible via digital catalogs and manufacturers' websites. In many cases, these data are not compatible with BIM evironments (e.g., they appear as PDF documents or written product specifications) and cannot be used directly in BIM models. The advancement of BIM software tools and online systems has provided more efficient means for practitioners to acquire and utilize building product information. In a BIM model, manufactured building products such as doors, windows, and masonry units can be defined as Building Object Models (BOM) (Eastman et al. 2011), also called building element or component models (Arnold and Wishart 2008, Costa and Madrazo 2015). BOMs are abstract representations of physical elements that have certain properties such as 2D or 3D geometric representations, geometry parameters, materials properties and representations, performance specifications, connection locations, and links to product distribution channels (Eastman et al. 2011, Knight 2012).

Specific BOMs developed by the manufacturers of building products are not directly included in BIM authoring tools such as Autodesk Revit, but are accessible from manufacturers' websites and online databases known as BOM libraries (Afsari and Eastman 2014). A BOM library or database is a collection of related data that is used to provide easy access, management, searching, and visualization of BIM objects. BOM libraries serve as access points for a variety of BIM objects, supporting hierarchical navigation, searches, downloads, and in some cases uploads of BOM files (Eastman et al. 2011, Costa and Madrazo 2015, Arnold and Wishart 2008). Having a consistent and standardized system of representing and storing BOM data for use in BIM models beginning at the early conceptual design stage would be especially beneficial in the later stages of building project development. Standardized object models would facilitate materials takeoff, cost estimation, and facility management in the later phases of a building project's development (AECMag 2014). BOM libraries provide designers with a wide selection of materials and comparison tools for effective component selection. In addition, they are beneficial for BIM projects because they enable the reuse of predefined models, reduce modeling costs, and cut total project time (Zhang and Xing 2013, AECMag 2014).

Generally speaking, BOMs represent geometry well. However, non-geometric data such as engineering properties, compatibility with other products, minimum order quantities, and lead times are usually missing. If data from a BOM library are downloaded into a model, it is possible that the data will become obsolete. Moreover, if data from a BOM library are linked or referenced, there is some risk that the links will become disconnected from the model. MUD was developed to address these limitations.

MUD acts as a special purpose BOM library for masonry unit products. This library is based on a proposed standardized system for classification and representation of masonry units, and tailored to act as a repository for masonry wall design and detailing software (Cavieres and Gentry 2015). MUD was intended to satisfy two main purposes: first, to act as a library of masonry units to be accessed via BIM software tools and embedded with information for design, engineering, and construction purposes; and second, to create an e-commerce platform for manufacturers, suppliers, masonry purchasers, and contractors to use when comparing, selling, and buying masonry construction materials. The data requirements for each of these functions is discussed below in greater detail.

### *5.1.4 MUD Development Process*

This research identified five steps required for the development of a comprehensive masonry unit database (see Figure 5-1) (Sharif and Gentry 2015a). The first was to map all of the project stakeholders and their activities throughout the course of a BIM-supported masonry building project’s lifecycle. Stakeholders and their tasks were represented in a process map that illustrated the different project tasks and exchange of information during the various stages of the building project. Based on the determined exchange requirements, the second step of MUD's development was to identify the specific set of masonry data and attributes for the building project's development and delivery. The masonry attributes were necessary for product classification and database organization.

The third step in MUD's development was selection of the database format, and the physical design, implementation, and instantiation of the database. The design needed to provide for storage of the data for all of the different types of masonry units in the structured system, so these data could be accessed easily and used by different project actors during the various stages of the project's development.

After development of MUD's back-end structure, the next two stages involved the design and implementation of the access structures for data import and export. This research presents two access structures for MUD. The first is a web-based front end with the ability to search, compare, and select masonry units. The second is a BIM plugin developed for Autodesk Revit with direct access to MUD, for use in selecting and representing masonry units in a BIM software environment.

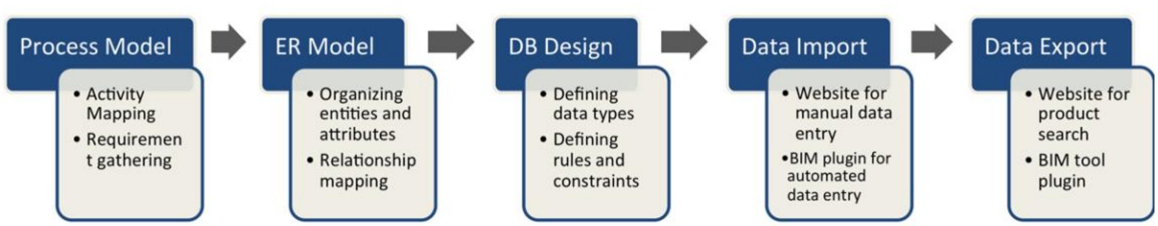


**Figure 5-1: MUD development process.**

## 5.2 Masonry Building Project Process Model

Building product information is required throughout the lifecycle of a building project, from design to construction and maintenance. The BIM-M research team completed a detailed analysis of a range of masonry design and construction projects, including a range of masonry assembly types such as veneer walls, load-bearing interior and exterior walls, and projects using custom cast and cut stone. Process and data models were developed based on the data obtained from these case studies (Lee et al. 2015).

The process models developed (also described as workflow) were used to represent activities related to both design and engineering, as well as project management and masonry supply. To classify and analyze the critical required masonry information needed to develop MUD, it was necessary to acquire a data model for representation and abstraction of the process; this allowed for the complexity of the data to be adequately reduced, enhancing our ability to focus on the most important information. We adopted a formal method for documenting these processes, using business process modeling notation (BPMN) (von Rosing et al. 2015). Elucidation of the data requirements of process models was first described by Eastman, Lee, and Sacks (Eastman, Lee, and Sacks 2002). In the past, BPMN has successfully been used to document the information requirements of the precast concrete and curtain wall industries (Jeong et al. 2009, Moya and Pons 2014).

The BPMN model was organized in a table format, with project actors listed in rows and project stages in columns. The interaction and information exchange among project actors and the flow of information from one stage of the project to the next were denoted as exchange models. The different masonry datapoints required for each exchange model were highlighted and identified for representation in MUD. To develop the process model for the masonry building project, project actors and stages were categorized based on the

OmniClass Construction Classification System (OCCS). This system is used to organize project information and develop electronic databases (OCCS 2015). Based on the OCCS definitions, each project stage was considered a "higher level of categorization of the principal segments of a project," while each "phase [was] a subordinate level of titling within a stage" and "disciplines [were] the practice areas and specialties of the actors (participants) that [carried] out the processes and procedures that [occurred] during the life cycle of [the] construction entity" (OCCS 2015). This research identified six main project disciplines (from OmniClass Table 33, "Disciplines") involved in six primary project development phases (from OmniClass Table 31, "Phases") (see Figure 5-2). The detailed process models are discussed in the BIM-M Phase II Project 1 report entitled "Masonry Unit Model Definition" (Gentry et al. 2014). A brief description of the data requirements for each project actor is discussed in this section.

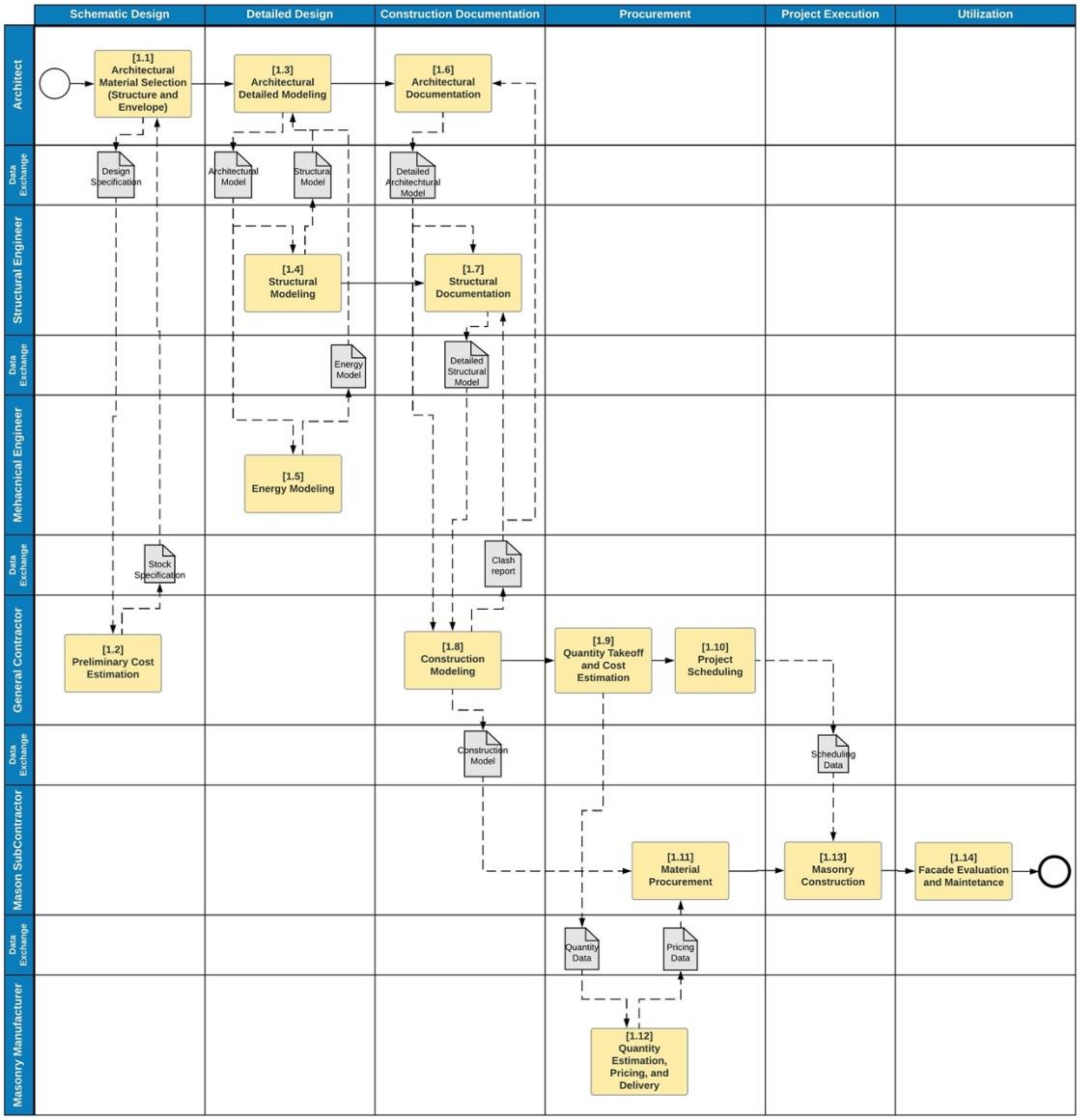


**Figure 5-2: Concise MUD BMPN model.**

### *5.2.1 MUD Data Requirements for Project actors*

The content and organization of the MUD schema was driven by an analysis of the data requirements of the major masonry industry stakeholders. In general, information requirements increase as a project proceeds, commensurate with the levels of development (LOD). As an example, MUD stores information on both generic and specific masonry unit types. While *generic* masonry units have sufficient information regarding unit shape, size, and conventional industry names, *specific* units are linked to the particular manufacturers and suppliers of each unit. These unit types are discussed in greater detail in Section 5.3.1. Below, the data requirements according to the main project actors at various stages of the building project are summarized.

**Architects:** In any BIM masonry building project, architects must be able to access masonry unit information in MUD, such as geometric, aesthetic (i.e., color and texture), and physical (e.g., thermal resistance) properties and the price of any masonry units selected.

In the schematic design stage, architects complete the selection of materials (see Figure 5-2) primarily based on the shape, color, and texture of the masonry units. They need access to an accurate tool for side-by-side color and texture comparison in 2D and 3D geometry, as well as images of the finished surface. In this stage, generic unit types are used to provide adequate information regarding masonry geometry. In the next step, the building materials selected can be used to produce 3D renderings of the conceptual design, with colors and textures applied to the building facade.

In the detailed design stage (see Figure 5-2), building floorplans and elevations are established. The exact locations of wall and door openings and aesthetic aspect of the façade treatment of those openings are determined. The architectural treatment of the building's corners, parapets, and transitions from non-masonry to masonry façade elements are finalized. Masonry units such as brick veneer, cast stone, and architectural CMUs (which have architectural implications) are selected by size, type, color, and manufacturer.

It is possible to enhance MUD in the future with manufacturers' and/or suppliers' information for specific unit data such as price, availability, and location of production. By accessing these details, architects will be able to determine if the selected masonry products meet the project's requirements during the early stages.

The construction documents stage entails the final specification of masonry materials and coordination of representations of the masonry units in the BIM models, schedules, and specifications. Vertical and horizontal sections are generated to show how the masonry interfaces with door and window jambs, headers, sills, and parapet elements. Finishing schedules (including masonry work) can be generated from the BIM model.

**Structural Engineers:** At the detailed design stage of a project, structural engineers must access the geometric, physical, and mechanical properties of masonry units in order to model, calculate, and analyze load-bearing structures. However, it should be noted that the structural capacity of masonry walls is determined from calculations related to the masonry wall assemblies and not the units themselves. In many cases, structural engineers must consider the unit properties in conjunction with the properties of allied materials (e.g., grout, mortar, rebar) to calculate the overall properties of masonry walls for specific axial, shear, and flexural strengths. A structural engineer assesses the efficacy of the gravity and lateral load systems and iterates with the architect to finalize a system solution. This may involve changing the strength, type, or size of the masonry units and/or global reconfiguration of load-bearing walls.

The structural construction documentation stage (see Figure 5-2) involves the detailing and specification of structural masonry. This can include gravity load-bearing elements, lateral load resistance structures, and veneer backup systems. For structural masonry, the provision of accessories compatible with the units or special masonry units such as bond beams or precast lintels takes place as the construction documents are assembled.

**Energy Analysts:** Energy analysts must access the geometric and physical property data of masonry units during the detailed design stage. This information is required to model and calculate the thermal characteristics of masonry walls from thermal resistivity, as well as surface characteristics and the density of the masonry units (see Figure 5-2). Energy models are used to establish conformance with state energy codes (typically based on a version of ASHRAE 90.1) (ASHRAE 2016) and determine the potential LEED points possible from the proposed design (US-GBC 2011). The results of the analysis with the suggested solutions are passed to the project architects for the editing and modification required for selection of the masonry units.

**General Contractors:** In the early stages of a project, the general contractor and design team are typically required to perform a preliminary cost estimation (see Figure 5-2) to determine whether the project is meeting the cost targets established early on in the design process. The goal of BIM-enabled cost estimation at this stage of a project is to identify the surface areas of masonry systems (Cheung et al. 2012) and apply square-foot costs. However, at this early project stage, only generic masonry units are used in the BIM model; it is not possible to link to the manufacturers of these products for cost estimation. Rather, an average price range for the selected generic units is used to generate early cost estimations.

In the construction documentation stage of a BIM masonry project, the general contractor is responsible for construction modeling and clash detection between the models provided by the architects and those developed by the project engineers (see Figure 5-2). After accessing the geometric information for the masonry units, the general contractor may return to the project actors to discuss possible clashes with the suggested replacement units, in order to produce the most effective results. In the next stage of the project, procurement, the general contractor can use the same masonry BIM model for project scheduling, as well as quantity takeoff and cost estimation. By including sufficient information in MUD, general contractors will be able to perform the required calculations for pricing, ordering, delivery, and erection schedule on the jobsite.

**Mason Contractors:** Mason contractors work in close collaboration with general contractors; however, they need to be able to access the complete range of masonry unit data during the procurement stage of the project. At this stage, if the BIM model developed by the architect contains generic masonry units, the mason contractor will be responsible for determining the appropriate specific masonry unit matches based on availability, price, and production location. At this stage, MUD can be used to prepare accurate cost estimations by replacing the generic units with the desired specific masonry units, facilitating electronic ordering and tracking from the materials' production plant to the job site.

**Masonry Manufacturers:** Masonry manufacturers are primarily producers of masonry units. They are in charge of providing the information about their masonry products that can then be authored into the database, based on the proposed guidelines. They are responsible for matching specific masonry units with the proper generic units, and providing supplementary information to make their products distinct on the market. However, information regarding the price and availability of a unit could be provided either by manufacturers or suppliers.

**Masonry Suppliers:** A masonry supplier is a vendor of masonry units who does not manufacture those units. Depending on the nature of the supply chain, a masonry supplier could provide relevant e-commerce information (e.g., price and availability) to MUD.

**Building Owners/Clients:** A building owner or client may be interested in reviewing the masonry materials selected by an architect at the schematic design stage; this mostly occurs in private residential projects. This stakeholder is usually primarily concerned with aesthetic features, along with price.

## 5.3 CLASSIFICATION OF MASONRY UNITS

The first step in the development of a BOM of masonry units for MUD was grouping and classification. A classification system of units allows for the comparison and selection of units with similar attributes. The attributes defined in our classification system must ultimately be machine readable by BIM systems. In this process, grouping and data abstraction played a significant role. The American Society of Testing and Materials (ASTM) describes classification as "a systematic arrangement or division of materials, products, systems, or services into groups based on similar characteristics such as origin, composition, properties, or use" (ASTM 2017). Class is a conceptual construct that identifies a collection of objects that have properties in common, while classification is a system of classes with specialized relationships (Afsari and Eastman 2016, Ekholm and Häggström 2011).

### 5.3.1 Masonry Material Classification

Based on the primary material and production system, masonry units can be classified into three major categories: clay brick, CMUs, and cast stone. CMUs are typically manufactured from a zero-slump concrete mix. These units are produced in a wide variety of colors, textures, shapes, configurations, and finishes (NCMA 1997). Units can be formed as solids or have hollow cores where rebar, grout, insulation, or plumbing and electrical chases can be placed. In the United States, the nominal dimensions of these units are usually in multiples of four for length and height (i.e., 4”, 8”, or 16”); they are generally in multiples of two for width (i.e., 4”, 6”, 8”, 10”, or 12”). The actual dimensions of the blocks are typically 3/8” smaller than the nominal size of a CMU, allowing for 3/8” mortar joints.

Clay bricks are produced by different manufacturing methods such as the stiff-mud (i.e., extrusion) and soft-mud processes; they come in various sizes, colors, and textures. Clay bricks are produced for two main functions, either architectural or structural (sometimes known as structural clay or thru-wall units). Architectural or facing bricks are used in veneer applications, or structurally in multi-wythe walls. In North America, these bricks are not commonly used in load-bearing applications, though they do carry their own weight and may help stiffen a backup wall section. Veneer brick is typically attached by ties to a backup system of CMUs, steel studs, concrete, or in some residential applications, wood studs. Structural clay units are generally larger than architectural bricks in order to allow for structural capacity in single-wythe applications. These bricks often have cores for reinforcing and grouting, similar to CMUs.

Cast stone is a custom (often parametric) precast masonry unit that resembles natural cut stone. Almost all cast stone is custom designed via a collaboration between the architect and cast stone producer, and used for building accent pieces such as lintels, sills, and trim. Because the range is quite variable, all pieces are generally made to order and require more complicated design drawings than would a standard masonry wall.

### *5.3.2 Generic vs. Specific Units*

Masonry units stored in MUD are identified as either generic or specific. Generic units are industry standard units sold by more than one manufacturer. They are most suitable for earlier design stages, when the geometry is specified but remaining properties (e.g., strength, color, texture) are as yet undetermined. The BOM of generic masonry units is defined based on the information provided by the masonry association and standards, with standardized unit sizes, geometric shapes, color families, and textures. However, generic units have no information regarding manufacturers or suppliers. They do not carry a stock keeping unit (SKU) or order code. Generic masonry units are used in general product selection and comparison by architects and engineers, as well as when creating the basis for modeling and representation of masonry units in BIM software tools as part of masonry wall systems.

Specific units are made by a particular manufacturer and can be purchased. In latter project development stages and as the masonry building model progresses, generic masonry units will be replaced by specific masonry units. In general, multiple specific units can be mapped to a single generic unit. Specific masonry units have more detailed information (e.g., updated geometry) based on the tolerance rate of the product, exact color and texture information, suppliers, price, and availability. As implemented, generic masonry units are open access in MUD; however, the information regarding specific units must be provided by the manufacturers, relying on their specific business model for accessibility of data (Afsari and Eastman 2014, Rundell 2008, AECMag 2014). Enhancing the building model with specific masonry unit information will provide the basis for e-commerce, including detailed product variations and specifications, selection of local products, availability, cost comparison and estimation, and product purchasing.

## 5.4 Database Development

The main part of MUD, as with any BOM library, is a database that stores all object data in a structured system. This can be accessed by different building project actors via BIM software tools and web portals at different stages of the project. Classifying building product models facilitates organization of building product libraries by providing standardized terminology and semantics (Afsari and Eastman 2016). In the present research, we propose a parametric and uniform classification and modeling system for masonry units, rather than a storage system for the BIM or CAD models provided by manufacturers. The manufacturer models currently accessible from websites and major BOM libraries are mostly non-parametric 3D objects. These BOMs are created with different BIM authoring tools in different formats and with no common procedure or standard for their creation (Costa and Madrazo 2015, Zhang and Xing 2013, AECMag 2014). However, the primarily parametric geometric nature of most masonry units (including CMUs and clay brick) provides the option for parametric definition and data storage of masonry units (as discussed in Section 5.4.3). For this purpose, a uniform classification and modeling system in the form of a relational database serves as a reasonable storage system for masonry unit data and related attributes. This database is accessible by both BIM software tools and web applications, offering the advantage of storing 2D and 3D CAD and BIM models provided by manufacturers.

### *5.4.1 Choice of database system*

The choice of a database system was mainly determined by the quantity of data to be carried and the target audience. In order to achieve a uniform data format and storing structure, we implemented a relational database system for MUD. Below we discuss the main reasons for this choice over other options, such as object-oriented and extensible markup language (XML) databases.

Relational databases provide data abstraction and application flexibility; they perform best with structured data. Consequently, if masonry unit data can be sufficiently organized into a coherently structured dataset, well-established relational database management systems (RDBMSs) such as MySQL, Microsoft SQL Server, and Oracle would provide high security and a reliable structure for the simultaneous access of many users from web and other access gates (Ramakrishnan and Gehrke 2000). High-level query languages such as Structured Query Language (SQL) provide an alternative to programming language interfaces. In addition, more recent versions of relational database systems such as the SQL standard (2008) for RDBMSs have incorporated many of the stronger features of object-relational databases (Navathe and Elmasri 2010). Conversely, some studies have suggested the use of XML (Fleming, Long, and Swindler 2012, Kong et al. 2005) and semantic web technologies such as Product Semantics Representation Language (Patil, Dutta, and Sriram 2005), Uniform Resource Identifier (Costa and

Madrazo 2015), and connection structures for IFC models like Object Information Pack (Nour 2010) for the development of MOD libraries. If MUD users require masonry data in other types of formats, there are existing methodologies for translating and mapping the structure of a relational schema into a semi-structured schema such as XML (Fong, Pang, and Bloor 2001).

### *5.4.2 Schema Design, EER Model, and Data Abstraction*

The main component of MUD is a database that stores all object data in a structured system. Based on the information gathered from stakeholders and domain experts and modeled in the process map, masonry unit information requirements were classified in a conceptual schema for use in database development. To design the conceptual data model for the database, the masonry unit information requirements gathered were then abstracted and classified. Conceptual modeling is an important phase in the design of a successful database application.

We acquired an enhanced entity relationship model (EER model) for representation of masonry unit data in MUD. The EER model's readability by nontechnical users was an important aspect of ensuring the complete identification of users' data requirements and reducing possible data conflicts.

Data abstraction is a fundamental process for conceptual model development. As Eastman has asserted, "an abstraction of some representation is a second representation in which details of the first are purposely omitted" (Eastman 1999). Traversing a hierarchy from top to bottom, a single term such as "masonry unit" was replaced with a set of terms such as "concrete masonry unit, architectural brick, structural brick, and cast stone."

Data abstraction helps with categorizing various masonry units of different shapes, materials, functionalities, and manufacturers. Despite all the variety, these units have certain properties in common that facilitated their classification in MUD. The main data required for MUD was categorized into *internal* data for generic masonry unit specification and *external* data for specific masonry unit specification. Internal data were classified as either geometry, material, physical properties, color, or texture. These sets of information are required for design, engineering, and construction purposes such as unit specification, comparison, and selection, satisfying the first goal of MUD. External unit information was categorized as manufacturers, suppliers, and building projects data, which are required for business activities such as cost estimation, availability queries, and unit of order verification, all of which are necessary if MUD is to act as an e-commerce platform. Each of these internal and external datapoints formed an entity type in the MUD EER model (see Figure 5-3).

### *5.4.3 Masonry Unit Attribute Determination*

The EER model developed for MUD is illustrated in Figure 5-3. Main entities and their associated attributes are discussed below.

*UNIT entity:* The core of the MUD schema model is the UNIT entity, which holds the core information about each masonry unit. All other entities such as GEOMETRY, MATERIAL, and MANUFACTURER have relationships such as *has, is_made_of,* and *is_made_by*. The attributes that define this entity are Globally Unique ID (GUID), name, type, family name, and nominal dimensions. UNIT entity, like all other entities in this model, has a GUID attribute for the unique identification of each entity in the entity set.

The name attribute denotes the commercial name that manufacturers specify for their masonry product. The type attribute classifies the masonry products at the highest level, based on their main material (which includes CMUs, clay brick, and cast stone masonry). The family name is used for grouping a set of masonry units with similar characteristics. Nominal dimensions are mostly used for the classification and grouping of masonry units. These nominal dimensions are supplemented by the actual dimensional attributes of the units in the GEOMETRY entity. The UNIT entity may also store (string) values for images or drawing file locations provided by masonry unit manufacturers.

*GEOMETRY entity:* Geometry is one the most important features of each unit; it has a direct impact on design and engineering tasks, as well as representation of the unit in BIM software environments. The attributes for this entity must be defined to satisfy the parametric data storage and 3D model generation of a wide range of common masonry units. Consequently, for MUD, the unit geometry was classified into three main groups: *regular* geometry, *special* geometry, and *custom* geometry.

Regular masonry units are produced by most masonry manufacturers, and come in almost identical sizes and shapes (within an acceptable range of tolerances). The geometry of these units can be categorized and defined to a high level of accuracy, based on their parametric attributes. From the values assigned, each masonry unit can be identically 3D-generated with the data stored in MUD. Both CMUs and clay brick have a substantial set of units that can be classified as regular. CMU general units have parent families such as stretchers, piers, corners, return corners, sashes, corner sashes, bond beams, conduits, lintels, open ends, headers, and starters, and subtype groups such as bullnosed, scored, and ribbed (e.g., circular, rectangular). Clay brick has two major categories, molded and extruded, with parent families including thin, face, structural, paver, etc.

Units with special geometry inherit most geometric attributes from regular unit geometry, though with some special features unique to these units. Units are usually produced by one specific manufacturer, based on their system of fabrication or particular preferences. It is possible to define a complete set of parametric attributes to represent these units. However, this set of unique attributes would add extensively to the complexity of the database data model and make it impractical in the physical development stage. As result, these units are represented with their closest counterparts from regular unit geometries and only marked as special so that manufacturers can provide more detailed information about each, based on user requirements.

Units with custom geometry are uniquely designed and produced based on the request of the project architect. Most cast and cut stone and some clay masonry are classified in this group. These units often have complex geometries that cannot be represented parametrically. The 3D representation of these units must be accomplished either with boundary representation (B-rep) or constructive solid geometry (CSG) models provided directly by the unit manufactures.

The attributes of the GEOMETRY entity are defined to adequately represent all of the units with regular geometry in the database and regenerate them in BIM applications. Parametric storage of geometric data (as compared to the storage of a 3D model of each unit) provides a very compact database, as well as an option for quick editing and updating of the unit information in both the database and BIM applications.

*COLOR entity:* The masonry unit color is the result of color ranges in the raw materials, aggregate mix, added coloring agents, or glaze color (in case of glazed brick). Color variations can be standard or special order. The attributes that define this entity are the color's name and family, which are assigned by the manufacturer. Based on the images of each masonry unit color provided by the manufacturer, the color of the unit is analyzed and classified into basic parametric attributes for color definition: red, green, and blue (RGB) color models and hue, saturation, and value (HSV). Krzywinski suggested a set of human-readable names for each color based on the range of RGB and HSV values; such names are beneficial for classification and comparison of masonry units in MUD (Krzywinski 2018).

*TEXTURE entity:* The texture of a masonry unit is an indicator of its appearance, feel, and consistency of surface. Texture can be defined as the pattern or configuration apparent in the exposed surface of a masonry unit. It applies to both clay and concrete masonry units, but the language used to describe it varies depending on the type of material. A name attribute is assigned by the manufacturer to a produced set of units. The family attribute is used for categorization of those units into more general texture groups, such as glazed, smooth, textured, and rough. For a parametric definition of texture, this research proposes attribute amplitude, a numerical scale ranging from 0 to 10 with zero referring to absolutely flat surfaces (e.g., glazed units) and 10 applying to units with rough textures (e.g., split or slumped faces).

*PHYSICAL_PROPERTY entity:* The physical properties of masonry units are required for engineering processes such as structural and energy analyses. Accordingly, this entity includes attributes for both the mechanical and thermal properties of masonry units. Some apply to both CMUs and clay brick, while the rest are only relevant to a certain type of masonry unit. Properties are determined based on ASTM standards and designed for engineering analysis at both the masonry unit and aggregate levels of wall assembly. The attributes of PHYSICAL_PROPERTY are listed in Table 1.

**Table 5-1 Attributes of the Physical Property Entity**

| Attribute | Applicable to | Attribute | Applicable to |
|---|---|---|---|
| **Thermal resistance** | CMUs and clay brick | Fire rating | CMUs and clay brick |
| **Solar reflectance** | CMUs and clay brick | Weight | CMUs and clay brick |
| **Density** | CMUs and clay brick | Compressive strength | CMUs and clay brick |
| **Modulus of elasticity** | CMUs and clay brick | Modulus of rigidity (diagonal tension or shear) | CMUs and clay brick |
| **Sound transmission Class** | CMUs and clay brick | Water absorption rate | Clay brick |
| **Cold absorption** | CMUs | Boiled absorption | Clay brick |
| **Initial rate of absorption** | Clay brick | Saturation coefficient | Clay brick |
| **Integral water repellent** | CMUs and clay brick | Efflorescence resistance | CMUs and clay brick |
| **Porosity** | CMUs and clay brick | Shrinkage coefficient | CMUs |
| **Coefficient of thermal expansion** | CMUs and clay brick | Creep coefficient | CMUs |

*MATERIAL entity:* Masonry units are made of a combination of different raw materials and through a variety of production processes. CMUs are made of a mixture of powdered Portland cement, water, sand, and gravel. Clay brick is comprised of natural clay (i.e., minerals such as kaolin and shale) and sand, and mixed with small amounts of additive components like manganese and barium for the production of color shades or improvement of chemical resistance (BIA 2006). The listing of materials and their percentages is important for projects with a LEED accreditation focus or when a building's embodied energy has to be minimized. The MATERIAL entity is defined by attributes including material name, type, source location, and percentage recycled.

*MANUFACTURER entity:* This entity associates the masonry unit with manufacturers of that specific product; the attributes are used to identify the company, including name, locations, and website. In addition, the “make” relationship between the UNIT and MANUFACTURER entities is elaborated upon through two additional attributes: cost and the availability of masonry units produced at that company

*SUPPLIER entity:* Masonry suppliers are vendors of masonry units. They link the masonry manufacturers with masonry contractors for purchasing purposes. The SUPPLIER entity in the MUD is defined by name, location(s), and website attributes. The relationship between this and the UNIT entity has additional attributes: cost and availability. The attributes assigned to the SUPPLIER entity and DISTRIBUTED_BY relationship are used in the comparison and selection of masonry suppliers based on their location, price, and stock availability. In addition, the SUPPLIER entity has an additional relationship, WORKS_WITH, which relates it to the MANUFACTURER entity.

*PROJECT entity:* This entity represents building projects for which custom masonry units have been designed and manufactured. Each project entity is defined by the following attributes: name of the project, owner of the project, and project location.

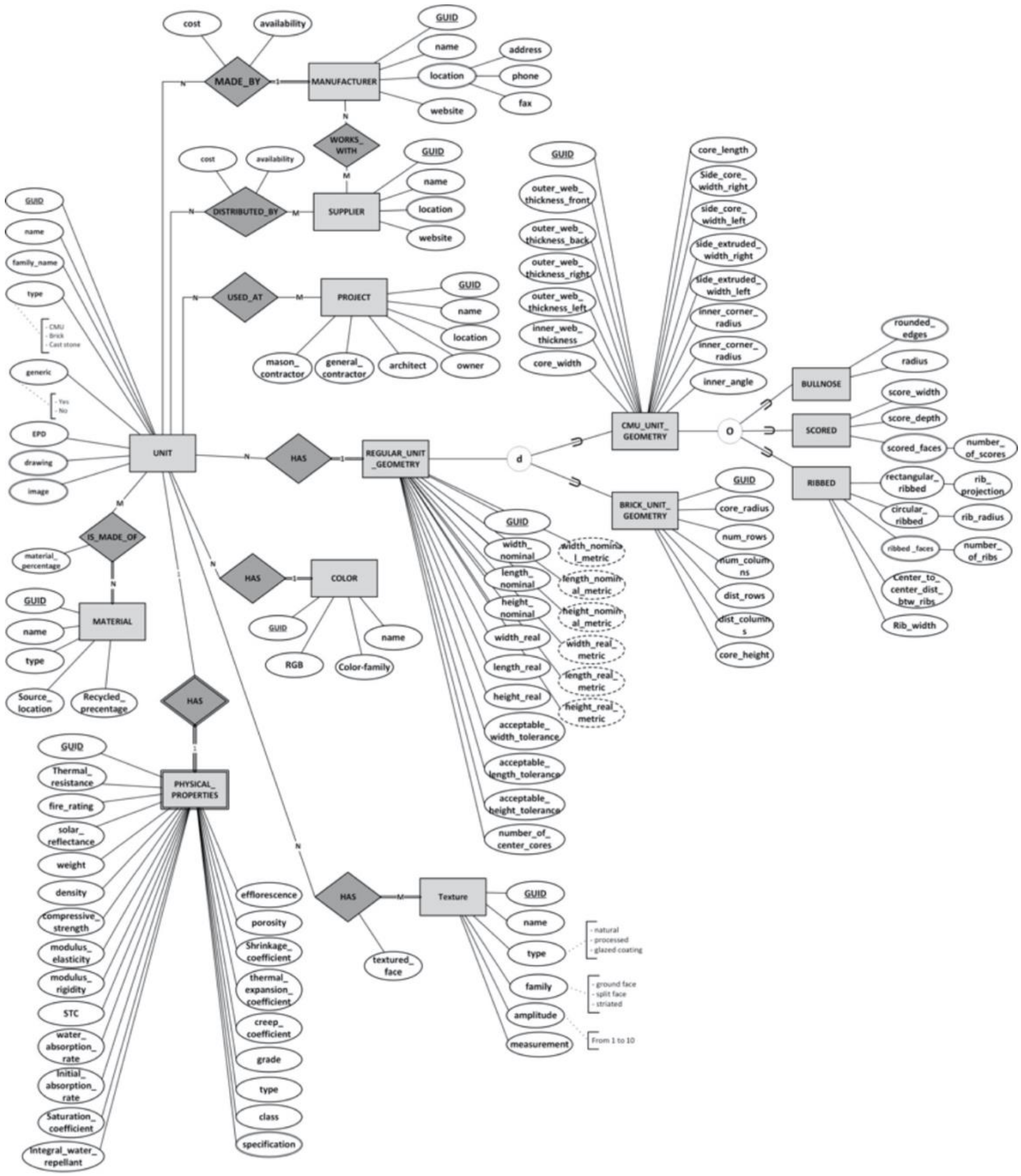


**Figure 5-3. Complete EER model of MUD, representing the main entities and attributes required for MUD's development.**

## 5.5 Data Model Mapping

The next stage of MUD's development was the actual implementation of the database by translating the developed ER model into SQL; the Microsoft SQL server was selected for this purpose. Here, the main goal was to convert the conceptual schema from a high-level data model (i.e., EER) into the implementation data model (Navathe and Elmasri 2010). RDBMS tools such as the Microsoft SQL server support and facilitate the storage, access, and modification of masonry unit data in an organized and efficient method. Data model mapping was followed by the physical design stage, in which the detailed data elements, types, and indexing options were specified. The MUD SQL model is composed of the following tables: Unit, Geometry, Material, Physical Properties, Color, Texture, Manufacturer, Supplier, and Project (see Figure 5-4).

To fulfill MUD's physical design, the database was initially instantiated with data for about 90 masonry units (i.e., clay brick and CMUs) from various manufacturers, in order to test the data requirements and relationships. As a result, the database data structure was edited to be able to incorporate all possible masonry unit requirement and data formats.

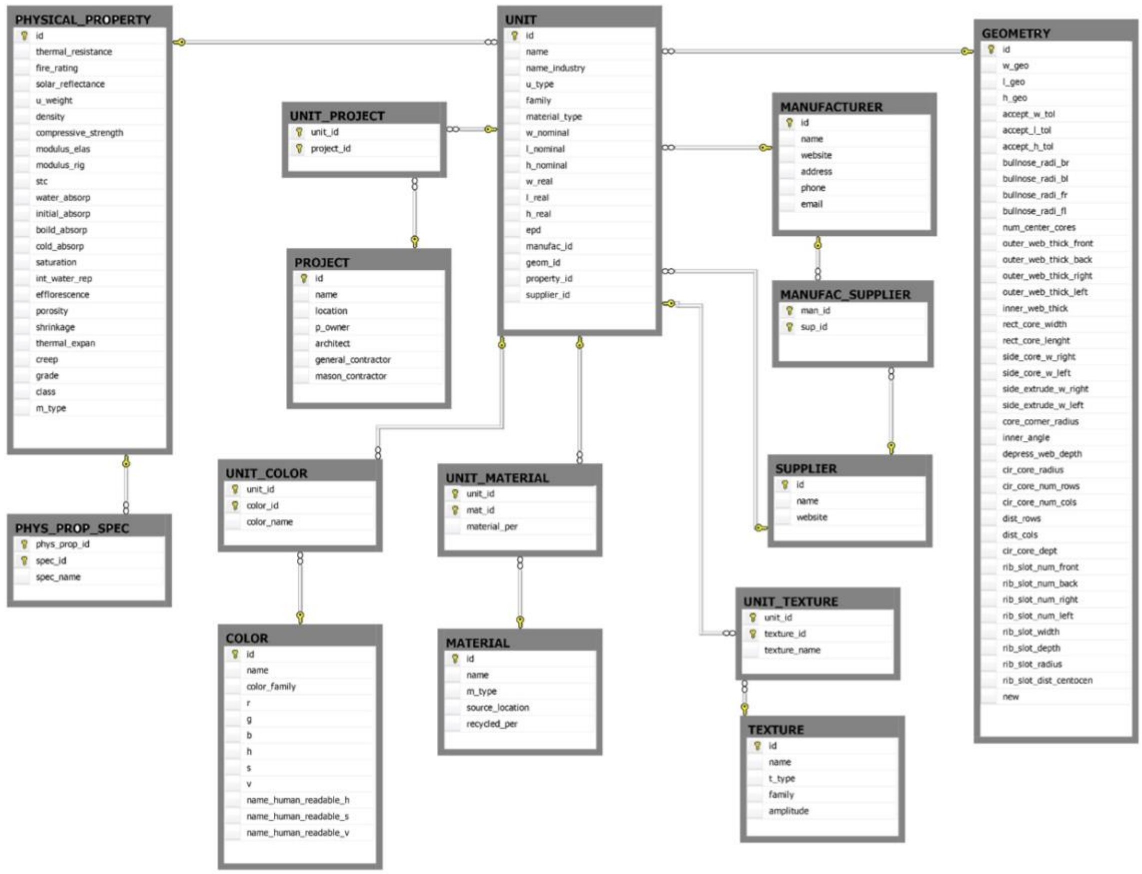


**Figure 5-4. MUD Microsoft SQL server interface showing the diagrams of the MUD entities generated and their relationships.**

### *5.5.1 Representing Generic and Specific Units*

MUD stores information about both generic and specific masonry units. In its initial development stage, the generic units were represented by information provided by the particular masonry association, with standard unit sizes, geometric shapes, color families, and textures; no information was included regarding the specific manufacturers of the units. Representation of generic units assists architects and engineers in general product selection and comparison, as well as in creating the basis for modeling and representation of masonry units in BIM software tools as part of masonry wall systems (Cavieres and Gentry 2015). It is anticipated that in future stages of development, the database will be enhanced with specific masonry unit information, including detailed data for manufacturers and suppliers of each masonry unit type. Specific unit information will provide the basis for e-commerce by making available detailed product variations and specifications, selection of local products, descriptions of availability, cost comparison and estimation, and finally product purchasing.

The features required for the representation of both generic and specific units is implemented in the MUD UNIT table with the attributes ‘generic_id’ and ‘manufacturer_id’ (foreign key to the Manufacturer table). In the case of a generic masonry unit, the values for both of these attributes have to be set to NULL, indicating that the entry has no reference to the MANUFACTURER table. Conversely, a specific masonry unit has an assigned value for the ‘generic_id’ that links to it a generic counterpart and an assigned value for the ‘manufacturer_id’, in order to link it to the proper manufacturer’s information in the database (see Figure 5-5). This representation method allows for the specific masonry unit to either inherit the base defined attribute values (e.g., geometry, physical properties) from their generic equivalents or assign an updated value explicit to the manufacturer. However, a complete set of constraints is required to specify the attributes that can be modified and their permitted range of change so that this model can still satisfy the replacement requirements for the generic equivalent.

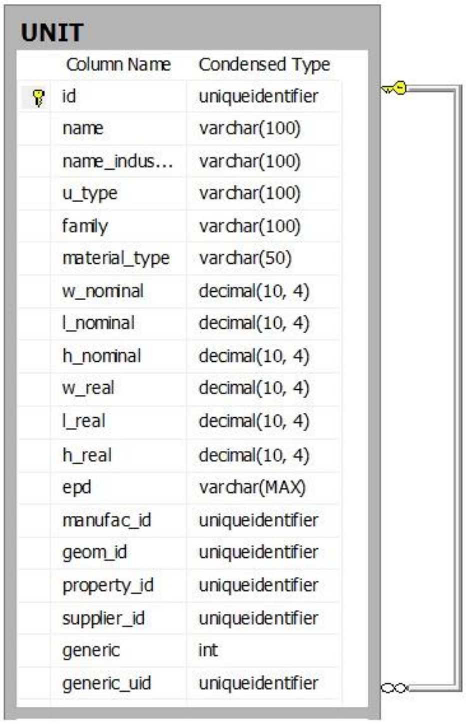


**Figure 5-5. UNIT entity table diagram in the Microsoft SQL server database for masonry units.**

## 5.6 Application Programs and Database Access

After initiation of the central RDBMSs, a structure for access and interaction with the database was developed. Most of the database interaction and data retrieval are executed through programs commonly known as application programs or database applications, and intended for use by end users (Navathe and Elmasri 2010). MUD application programs are intended to be used by trade associates, masonry product manufacturers, and materials suppliers for data input to the database, as well as by designers, engineers, and contractors for data access and use in BIM applications. These program applications contain units that use SQL statements to manipulate records in the database. For this purpose, the application program accepts parametric input values and outputs relevant SQL statements for the intended manipulation procedure for the database instance (Chan, Cheung, and Tse 2005). Both the data input and output application programs for MUD can implement web interfaces and plugins for BIM platforms, where they have direct access to the database for data manipulation (see Figure 5-6).

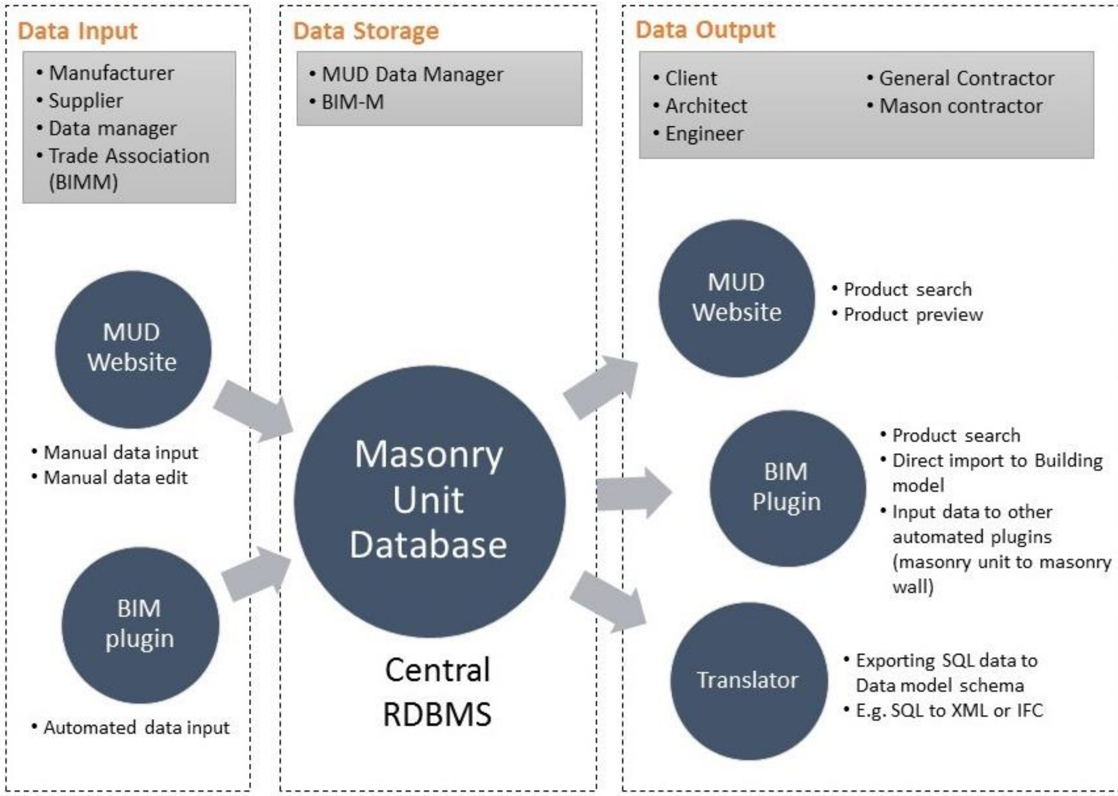


**Figure 5-6. MUD and data import and export structures.**

### *5.6.1 MUD Data Input*

The data stored in MUD require a unified representation system, which is necessary for effective and coherent access to masonry unit information. Therefore, it was necessary that producers and suppliers of masonry products provide a unified set of data regarding their products. This was achieved by providing tools for implementing the classifications and standards defined as MUD parameters and attributes. The data input structure for MUD provides a system for validating the model data against predefined parameters before accepting and storing datapoints in the database (Zhang and Xing 2013). This standardized system for both generic and specific masonry units allows masonry designers, masonry contractors, and product purchasers to easily compare and select their preferred units.

For MUD, two automated systems of data import were envisioned: first, a web-based frontend with a spreadsheet format that has direct connection to the database, and second, a plugin for BIM platforms that can import common 3D models of masonry units in common CAD formats and automatically extract geometric information. The extracted geometric information is transferred to MUD's SQL database, though manual information input for other attributes (such as color and texture) is required.

### *5.6.2 MUD Data Output*

After the compilation of new and updating of existing data on masonry units by masonry domain experts for generic units and product manufacturers and suppliers for specific units, MUD users are able to access information for masonry project development and embedment in their BIM models. As discussed by Zhang and Xing, BOM libraries require a navigation system to support multiple classifications to find object models and support user annotated tags for customization (Zhang and Xing 2013). Designers browse product and materials options to compare them, and thus need to visualize those products and materials (Ofluoglu, Coyne, and Lee 2002). This task, traditionally performed via manually constructed sample boards, can also be achieved in the digital MUD environment. Using the same method as is employed for data input, MUD data are accessed via two main access points: a web-based access platform for searching, viewing, and comparing units, and BIM plugins for direct access and use of masonry BOMs in BIM platforms such as Autodesk Revit.

A web-based application program for representation of MUD data facilitates quick access, searches, and the comparison of masonry units for different types of users. The website provides representations of masonry units via 3D, color, and texture images. In addition, digitally tagged masonry units are searchable on the web platform through keywords, as well as via category selection menus. The second option for exporting data from MUD is the direct transportation of inquiry results from the database into a BIM tool such as Autodesk Revit in the form of 3D geometric models.

### *5.6.3 MUD Validation Application*

As part of phase II of the BIM-M initiative, a database application was developed to validate the adequacy of the MUD schema. Autodesk Dynamo was acquired as a special plugin developer for BIM, and specifically for Revit. The developed Dynamo plugin was based on a Python script with the ability to connect to SQL management systems and run SQL queries to import stored dimensions and attributes from the database into the Dynamo environment, generate 3D models based on the imported data, and export the geometry as a family object in the Revit architecture (see Figure 5-7). With this plugin, the main entity used for the 3D parametric generation of masonry units is the geometry entity in MUD. This entity contains attributes for the parametric representation of both CMUs and clay brick masonry units (see Figure 5-8). Different data entries for attributes in each query for a specific masonry unit, along with the conditional rules defined in both SQL and Python, result in the on the fly generation of each single unit in the Revit environment.

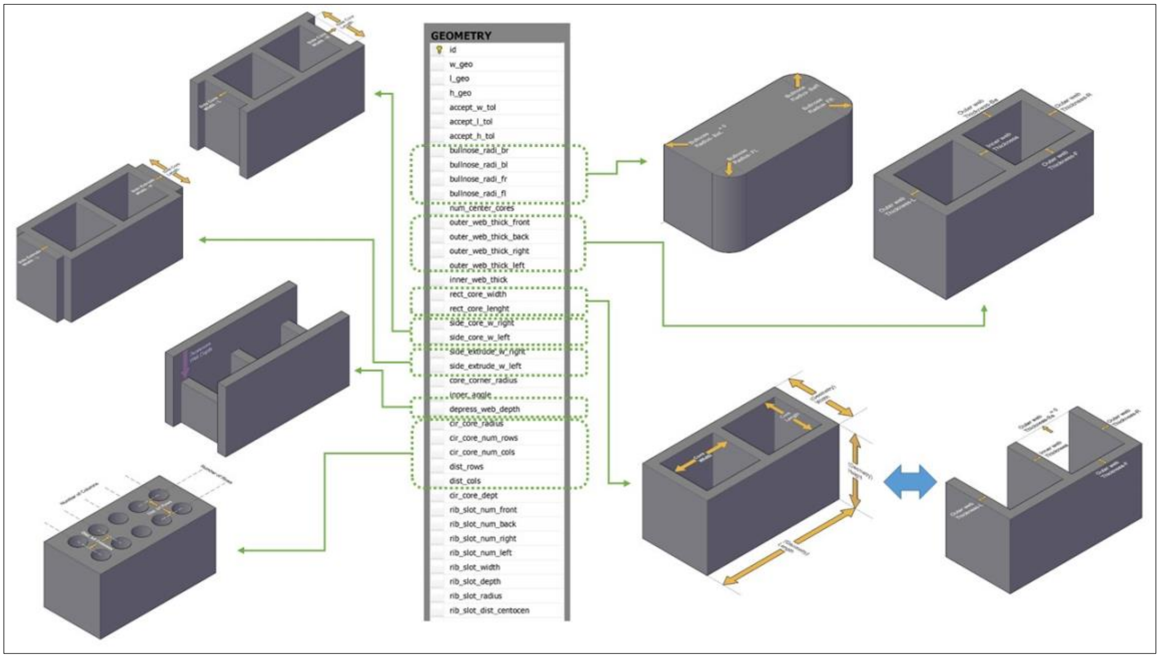


**Figure 5-7.Attributes defined in MUD for parametric geometric definition of both clay brick and CMU masonry modules.**

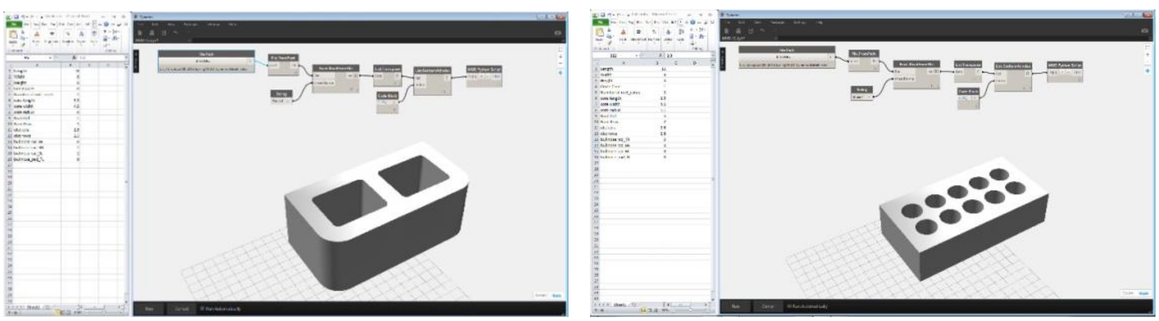

**Figure 5-8. MUD masonry unit 3D generator in the Revit Dynamo environment; two different CMU and clay modules were generated based on different MUD queries.**

### *5.6.4 MUD Commercial Applications*

The BIM-M initiative selected the CAD Technology Center (CTC) (https://cadtechnologycenter.com/) to be the commercialization partner for implementation of MUD. The research team continued to provide technical oversight.

The primary components of the commercialized MUD Version 1 (MUD V1) include: (1) a relational database schema adapted from this research and implemented by CTC, used to store masonry unit information; (2) masonry unit input web pages for entering masonry unit data into MUD, which can only be accessed by approved members; (3) a user web portal that provides access to MUD, including the ability to browse masonry units and download BIM and CAD files (http://www.mudb.org); (4) a Revit application that generates masonry units from parameters stored in MUD and allows for direct import of masonry units into Revit as families; and (5) a set of generic masonry units from the concrete, structural clay, and veneer clay masonry families populated into the database.

MUD V1 includes a selection of masonry units from the concrete, structural clay, and veneer clay masonry industries. These elements represent the largest segments in the industry. More than 100 unique unit geometries were identified for inclusion in MUD V1. Figure 5-9 depicts the 8-inch CMUs slated for inclusion.

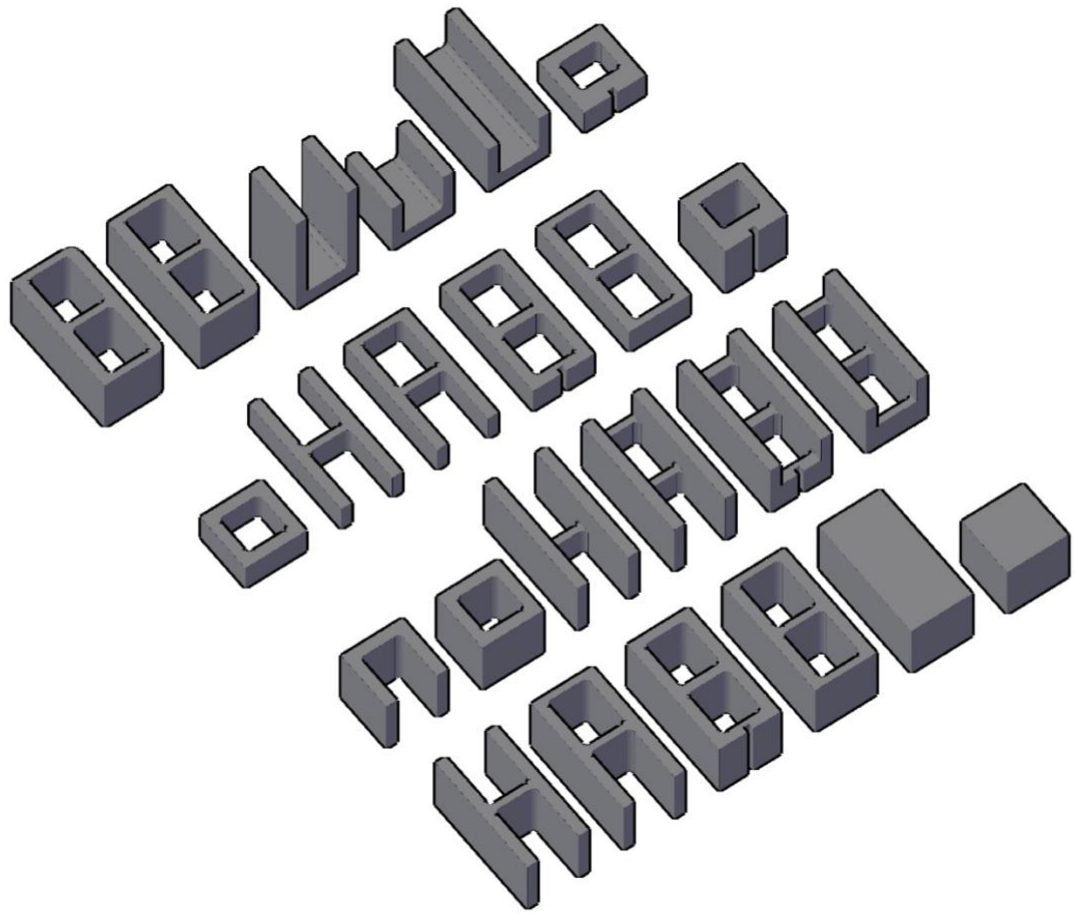

**Figure 5-9. 8-inch CMUs identified by NCMA for inclusion in MUD.**

The web portal for MUD was developed to facilitate user access to the database (see Figure 5-11). The portal groups masonry units by material type and size (see Figure 5-10). In future versions, adhered veneers, cast and natural stone, tile, and thin brick will also be added.

MUD V1 contains generic units only. Version 2 (MUD V2) will add masonry units from additional industry segments, including: tile, cut stone, and manufactured stone. Thus, it will extend MUD to support non-parametric geometries. It is anticipated that Version 3 (MUD V3) will contain specific units. Generic and specific masonry units stored in MUD will have either parametric or custom geometric shapes. Units with parametric geometric shapes are generated from a set of shape parameters stored in the geometry table in the database (see Figure 5-7). Conversely, masonry units with custom geometric shapes have complex features and are difficult to represent in tabular form. These units will be explicitly modeled in multiple CAD and BIM applications, and then stored in the database for download.

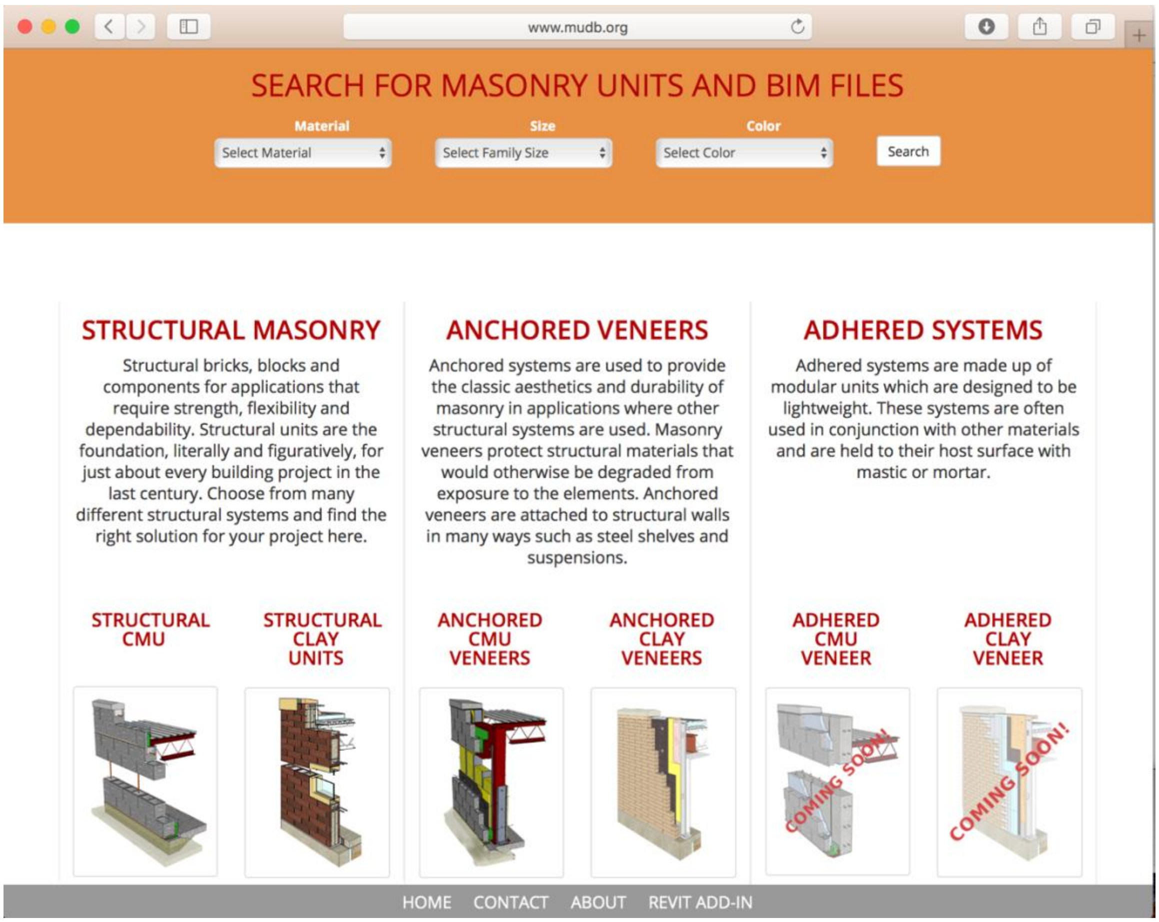


**Figure 5-10. MUD web portal home page.**

In addition to the MUD web portal, the Masonry Unit Generator add-on for Autodesk Revit was developed to allow for direct access to MUD through the Revit plugin. The plugin generates the units parametrically and embeds them in the Revit model as Revit families (see Figure 5-11). The Revit plugin accesses the MUD database stored on CTC servers, ensuring that the most current version of the data is embedded in Revit. It features graphic representations of units, search capability, and can save native Revit families to a user's materials library for later use.

The metadata stored in MUD provides for enhanced search options in both the MUD web portal and Masonry Unit Generator add-on. On the add-on search tab, there are several controls to aid with filtering and refining the search results (see Figure 5-12). The text search is enhanced with the ability to query the database for matching terms (in addition to the typed term). There is also the option to search the database based on various criteria such as material (e.g., extruded fired clay, lightweight concrete, medium weight concrete, normal weight concrete), standard masonry unit sizes as length measures (i.e., 4", 6", 8", 10", 12"), traditional names (i.e., King, Meridian, Modular, Norman, and Queen), and colors (e.g., brown, black, gray, etc. in dark, medium, and light shades). In the next step, the selected masonry unit instance can be directly inserted into a project, similar to the Revit functionality for inserting family types from the project browser (see Figure 5-11). Additionally, masonry unit models can be generated and saved directly to folders on the user workstation for future distribution and use.

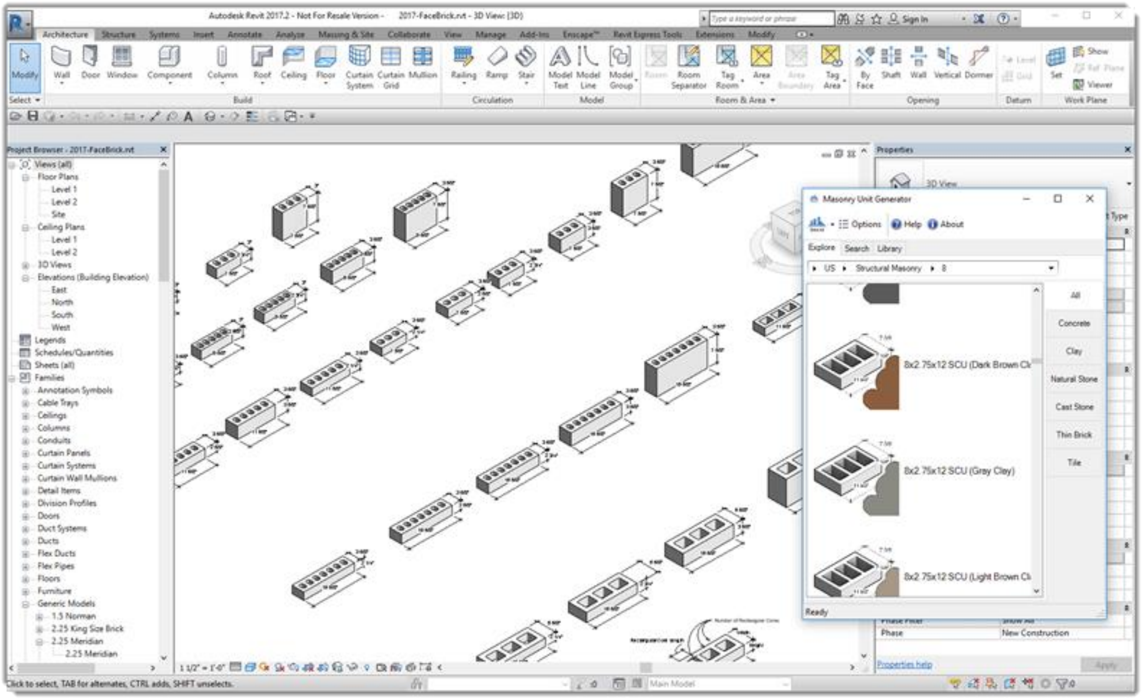

**Figure 5-11. Access to MUD via the Revit plugin.**

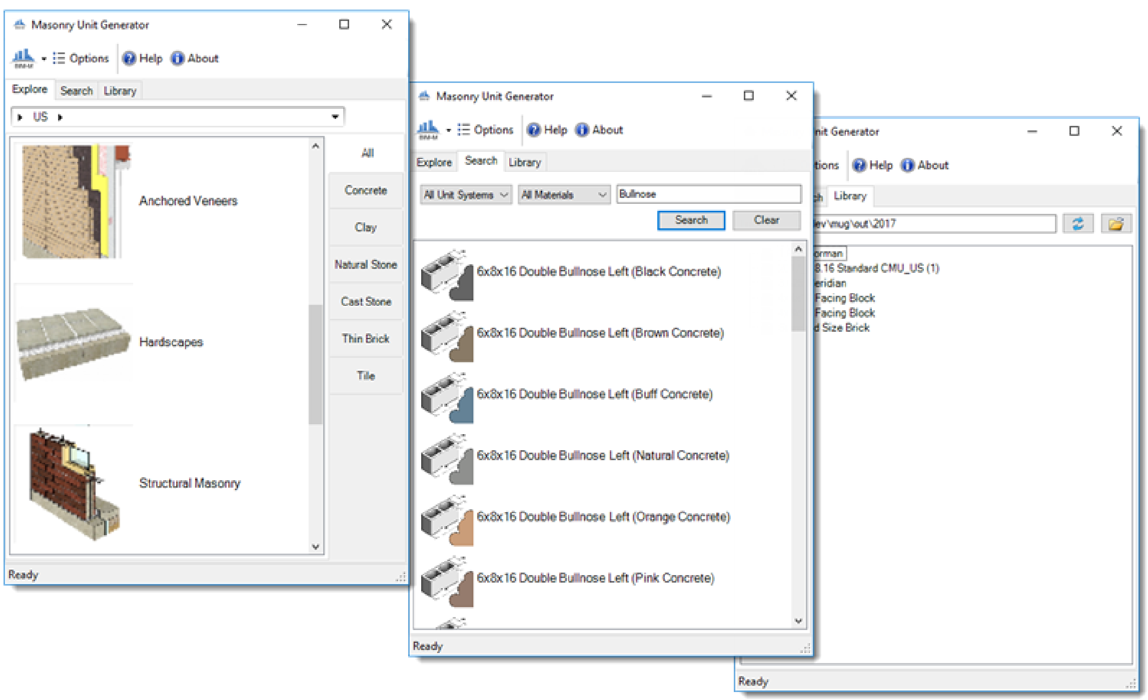


**Figure 5-12. Search options for masonry units in the MUD Revit plugin.**

## 5.7 Future Steps towards BOM Implementation

The release of the MUD V1 commercial application, as discussed in Section 5.6.4, will provide the chance for users to engage with the database. Collecting meaningful customer feedback requires an extended use of the application by a range of users such as architects, engineers, general contractors, and masonry subcontractors. Reviewing their feedback will provide us with valuable information for MUD V2. A number of additions for MUD V2 have already been anticipated by researchers at the Georgia Institute of Technology, and additional modifications have been proposed by industry stakeholders as part of BIM-M community outreach through developers' workshops and at BIM-M symposiums. This section outlines the specifications for improving the functionality of MUD, adding additional units, and preparing for long-term development and maintenance by the masonry industry once the BIM-M initiative has ended.

### *5.7.1 Additional Masonry Unit Types*

At the current stage, MUD only represents masonry units with regular geometries, including clay brick and CMUs. However, MUD should have the ability to incorporate all ranges of masonry units with custom and one-of-a-kind geometric shapes, including cut and cast stone. BIM-M has commitments from the manufactured stone (through the NCMA) and dimensioned granite stone industries for MUD V2. The International Masonry Institute will also support the addition of tile into MUD. Thus, the MUD schema must be extended to support these unit types.

### *5.7.2 Colors and Textures*

The colors and textures of masonry units have only been partially implemented in MUD V1. The proposed schema for MUD includes the potential to store bitmapped images of real masonry surfaces, so that the colors and textures can be mapped onto the units when viewed and downloaded. One of the difficulties of this approach is that handling colors and textures depends on CAD platforms such as AutoCAD and Revit. Consequently, a unified and platform-independent approach will be modified as a part of MUD V2, in preparation for hosting manufacturer-specific units in MUD V3.

### *5.7.3 MUD Plugins for Masonry Software Tools*

There are existing software tools specific to masonry design and quantity take-off, such as CAD BLOX (http://www.cadblox.com) and Tradesmen's Software (https://tradesmens.com). Feedback from stakeholders identified a need for the ability to access MUD via such software. This export requires that the database be version controlled and exported in a shared data format. In is anticipated that in close coordination with primary stakeholders Tradesmen's Software and CAD BLOX, MUD V2 will implement the proper export format.

### *5.7.4 Implementation of Specific Units*

A system for managing the association between generic units provided by industry trade associations and specific units sold by manufacturers still needs to be implemented. In this way, MUD will contain a rich assortment of masonry units, including geometries (with both color and texture), which can then be mapped to specific units sold by industry participants. These additions are anticipated for MUD V3.

### *5.7.5 Incorporation of MUD Entities in BIM Wall Assemblies*

The main outcome of MUD in a BIM environment will be the incorporation of masonry unit geometry models at different LOD into BIM masonry wall assemblies, offering a variety of masonry units and arrangements. While at LOD 100 and 200 these data models can be used by architects in BIM architectural models, LOD 300 and 400 will specify detailed fabrication-level models and shop drawings used by masonry subcontractors. The propagation of individual masonry units into the BIM model, if required, will occur at LOD 400. The masonry wall schema concept should support the selective placement of masonry units into the model on a region-by-region basis. In this approach, the masonry units in MUD will be parametrically generated within a specified local coordinate system, allowing them to be merged with the masonry hatch pattern generated at lower LOD levels at either the wall's face or centerline, as appropriate (Gentry et al. 2016). This is an important aspect of Phases II and III of masonry wall and BIM-M specification projects.

## 5.8 Summary and Conclusions

The semantics of masonry walls are largely missing from current BIM applications. To integrate masonry into BIM, definitions for masonry units and walls that are relevant in the physical (as well as computational) world are needed. The data models for masonry units, as described in this research, forms the basis for the computational representation of masonry in BIM. The geometry of most clay and concrete masonry units can be represented parametrically in a relational database. The geometric data can then be joined relationally with data regarding engineering properties, color, and texture, so that the masonry units can be populated into architectural, structural, and construction BIM models.

The academic research and commercial development of this masonry unit database has been one of the most successful projects within BIM-M. It is evident that purchasers and specifiers of such units require more immediate access to masonry unit data. It is also clear from recent BIM-M symposiums that masonry stakeholders are eager for unified information regarding the geometry and properties of the masonry units used to develop MUD as a platform for masonry design and e-commerce.

# CHAPTER 6. REINFORCED COMPOSITE BRICK MASONRY FOR OFFSITE ROBOTIC FABRICATION

**Abstract**

Masonry construction is a highly repetitive and labor-intensive process. Industrial robots are now widely used for materializing complex and non-standard geometrical forms in architecture, including masonry structures. However, adopting robotic arms for onsite construction faces challenges such as scale, mobility, and environmental conditions. In this research, we introduce a new composite construction technique for reinforced masonry as an alternative to traditional wet or dry stacking. Our composite brick structure supports offsite fabrication, including load-bearing structural stability, transportability, and assembly. We also discuss an algorithmic solution for testing the constructability of various complex wall forms for the composite system. We identified four rules to ensure the constructability and structural integrity of this type of wall structure. These rules were translated into a set of criteria via a Grasshopper plugin we developed for this purpose. By using this plugin, designers can test their brick assemblies with curved surface forms for structural stability and robotic constructability. We conclude by presenting three robotically fabricated masonry wall prototypes as proof of concept.

## 6.1 Introduction

Robots are well-suited for tasks that require high accuracy, repetition, and strength. In the building construction domain, numerous studies have demonstrated the exceptional capabilities of industrial robots, not only for fabricating geometrically complicated components but also in assembling complex structures. These fundamental qualities have made robots an excellent tool for masonry construction, a highly repetitive and labor-intensive processes that requires a substantial workforce. Masonry units have simple orthogonal shapes that make them compatible with and easy to handle by robotic grip systems.

In the late 1980s, a time when computers and robots were becoming more accessible, masonry construction quickly became one of the first targets of construction research. At this time, researchers reporting on this topic mostly addressed the development of new robotic masonry fabrication systems that could operate in controlled indoor facilities. Anliker developed one of the earliest implementations of a machine that could produce prefabricated brick walls in any length up to 8 m (Anliker 1988). Lehtinen et al. developed two masonry robotic systems that used adhesive material instead of traditional mortar. They also considered the economics and feasibly of such systems as compared to traditional manual masonry laying (Lehtinen, Salo, and Aatlo 1989). Bernold et al. developed a control system for a prototype bricklaying robot that took advantage of the real-time integration of actuators and sensors (Bernold, Altobelli, and Taylor 1992).

This first generation of masonry robotics research focused on how best to take advantage of the power and accuracy of these new automated machines in order to increase productivity and reduce associated costs. In most cases, the actual operation and simple brick patterns were adopted directly from manual masonry work. The robotic systems developed were highly specialized and expensive, and could not deal with unpredictable and unique construction site conditions. Thus, they never became commonplace in construction. In the late 1990s and early 2000s, there was tremendous improvement in industrial robot technology. This new generation of robots was fast, powerful, highly accurate, and reliable. Wide adoption in many industrial applications made them commercially affordable and accessible by a broader set of users (Bechthold 2010).

The second generation of studies on the application of robotics in architecture and construction adopted commercially available industrial robotic arms, and instead of hardware development, focused on materializing complex and non-standard geometrical forms. Brick units were the first material leveraged to exhibit the types of non-orthogonal structures that could be constructed. The Gramazio and Kohler research group produced a series of brick walls with double-curved surfaces in which the placement of each brick was unique. Based on the parametrized program they developed, the robot precisely laid individual bricks at the desired angle and position (Bonwetsch et al. 2006, Bonwetsch 2012, Bonwetsch, Bärtschi, and Helmreich 2013, Bonwetsch 2015). These studies demonstrated the powerful capabilities of industrial robots when combined with programming for creative design.

Now, the next step for the researchers is to provide solutions and the infrastructure required to make masonry robotics applicable to actual building projects. The first approach considered taking robots out of the controlled lab environment, addressing the requirements for onsite construction. Helm et al. examined how existing industrial robotic arms could be equipped with mobile bases to assist with maneuvering at the site. Their approach demonstrated how the basic capabilities of robotic arms could be extended, rather than investing in the development of new machinery (Helm et al. 2012). Research by Feng et al. and Dorfler et al. presented various vision and scanning methods to enhance mobile robots, making them able to navigate unstructured worksites and handle material dimension tolerances (Feng et al. 2014, Dörfler et al. 2016). Finally, Construction Robotics developed SAM, the first commercialized version of a mobile robot for onsite masonry construction (Peters and Belden 2014).

However, with all the advances in onsite masonry robotics, the main challenge remains repurposing robots that were initially intended to serve in industries like automotive manufacturing. Adopting these robots for construction faces challenges such as scale, mobility, and environmental conditions. Robotic arms are designed to be stationery and work in confined cells, handling objects they can encompass. Even when robots are equipped with a rail system, the range of movement is entirely predefined and calculated (Bonwetsch 2015). These prerequisites for proper robot functionality are at odds with onsite building construction situations. Building structures are many times larger than most robotic arms, and building components are heavier than the average robot's payload. Even if the robots are equipped with mobile bases, they are still confined to ground level at construction sites, require complex vision and navigation integration, and can only perform in ideal weather conditions in terms of temperature and humidity. The chance of broad adoption for onsite construction remains slim unless the new generation of robots is designed for a broader range of construction conditions (Bruckmann et al. 2016).

An alternative solution for using current industrial robots for construction purposes is to improve the offsite robotic prefabrication of building components. Offsite prefabrication takes advantage of lean production concepts and is regarded as an effective project delivery approach (Albus 2018, Smith and Quale 2017). With proper fabrication methods, robots can perform effectively in a controlled production environment. Then, the building components produced can be transported to the site for assembly and installation.

To achieve this goal, the masonry structures must satisfy structural stability requirements (Cavieres, Gentry, and Al-Haddad 2011) and address transportation and onsite assembly considerations. In this research, we introduce a novel high-performance reinforced composite brick masonry system (Biggs 2016).

We propose a composite construction technique that can serve as an alternative to traditional wet or dry stacking. Dry stacking, which in some cases is enhanced for robotic masonry by adhesive materials (Bärtschi et al. 2010, Bonwetsch 2012), has demonstrated only limited structural performance for construction purposes. Traditional wet mortar masonry (Xu, Luo, and Gao 2019, Peters and Belden 2014) also faces problems such as difficulties with controlling the thickness of the mortar mixture in changing environmental conditions. Our composite brick structure fits the criteria for offsite robotic fabrication, including load-bearing structural stability, transportability, and assembly. In the following sections, we introduce our reinforced composite masonry structure, discuss an algorithmic solution that allows for testing various complex wall forms for constructability, and present three robotically fabricated masonry wall prototypes.

## 6.2 High-Performance Reinforced Composite Brick Masonry

In architectural projects, brick masonry can be used to make either load-bearing and self-supporting structures (Hendry 1998) or thin veneer walls supported by a building's primary structure (usually for building enclosure and aesthetic purposes) (Liang and Memari 2011). Structural applications mainly rely on masonry's inherent ability to tolerate compressive stresses. However, purely compressive masonry with no tensile reinforcement can limit design options to either thick walls that use tremendous amounts of material, or thin structures with bespoke force-derived forms such as thin tile vaults (Trubiano, Dessi-Olive, and Gentry 2019).

In this project, we adopted a technique previously developed in a funded research project at the Georgia Institute of Technology, focused on the construction of prefabricated veneer walls. This technique targets high-performance self-supporting brick masonry walls using the principles of composite construction and is especially suited to robotic prefabrication. This new reinforced masonry-based structural system expands masonry's potential for use in the presence of tension. This masonry and reinforcement system has three main elements: three-core bricks, small-diameter high-strength rods passing through the brick cores, and a unique mixture of grout poured into the cores that acts as a composite.

The first test project for this new reinforcement scheme addressed the production of conventional flat brick masonry walls. While this wall was built by masons in a factory setting, it was strong enough to lift and be transported from the shop floor to the construction site. To demonstrate the technology, a reinforced masonry cantilever was built using glass fiber reinforced polymer (GFRP) bars for tensile reinforcement. The relatively low elastic modulus of the 6 mm diameter bars meant they could be threaded between the sliding and shifting apertures in the three-core brick pattern. Though flexible relative to conventional steel bars, GFRP bars have tremendous tensile strength (around 800 MPa) and in this case acted as thread-like stitches binding the unit bricks together. The grout also served as a composite.  A self-consolidating ultra-high-performance cement (UHPC) grout mixed with filaments of 25 mm cold-drawn stainless steel fiber at a 1% volume fraction facilitated the bonding of the GFRP bars to the brick cells. The steel fiber grout dramatically increased the bond strength between the grout and rebar and allowed for decreased lap-splice lengths between bars. Finally, a conventional 9-gauge steel joint reinforcement was used in the continuous joints to provide strength across the width of the walls (see Figure 6-1).

**Figure 6-1. Manual fabrication test of reinforced brick wall with GFRP bars: (a) UHPC grout with steel fiber, (b) bond beam at the top of the walls used to attach lifting anchors and spread lifting loads across the walls, and (c) a small-scale wall being lifted by a forklift.**

An adaptation of this technique was used for the prefabricated walls in our robotic fabrication workshop. Bricks were laid in a dry stack bond with the aid of a robotic arm. Instead of GFRP bars, 6 mm high-strength steel-threaded rods were used as reinforcement to bind the horizontal brick courses together. These modifications enabled the generation of non-orthogonal and complex geometric shapes and added to the structural stability and out-of-plane strength of the walls (see Figure 6-2).

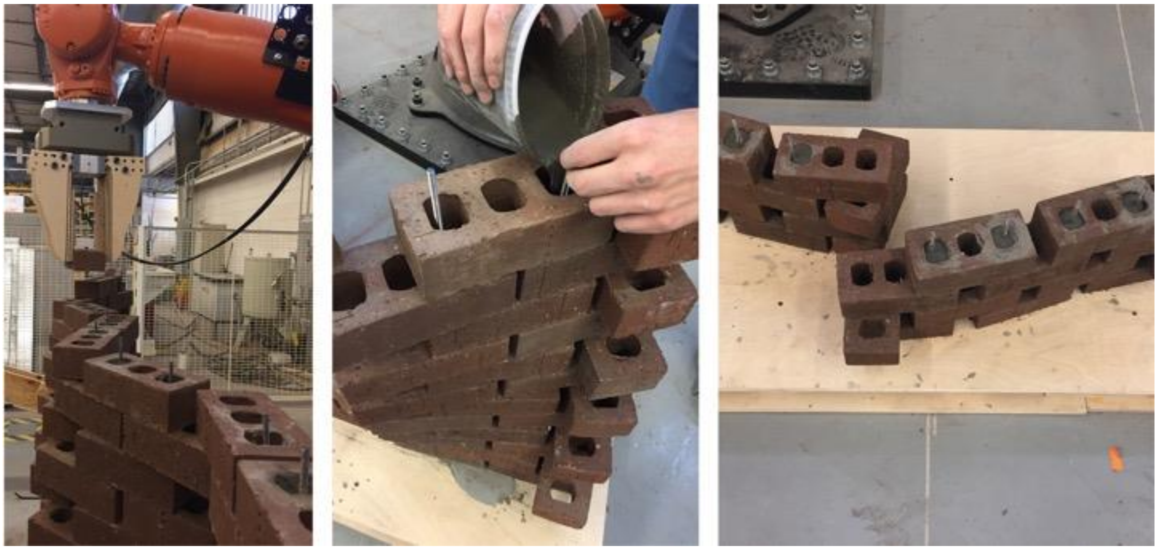

**Figure 6-2. Reinforced brick wall with steel-threaded rods for the robotic fabrication test.**

## 6.3 Algorithmic Constructability Analysis

The composite reinforced brick structure developed for the present research had two main features that made it different from masonry structures that only work with compressive strength. First, the tensile strength of the masonry wall system provided the opportunity for the design and fabrication of complex geometric forms not possible with purely compressive structures. Second, the composite brick had the structural strength required to safely and securely lift and transport the prefabricated wall panels to the construction site. These composite masonry walls could take many geometric forms. In this research, we identified four rules to ensure the constructability and structural integrity of this type of wall structure for various geometric forms.

Following is the logic for each of the rules and their requirements. These rules were translated into a set of algorithmic criteria via a Grasshopper plugin we developed for this purpose. By using this plugin, designers can test their brick assemblies with curved surface forms for structural stability and robotic constructability. This application works as a complementary module for the KUKAprc Grasshopper plugin used for simulation and the programming of KUKA robots. In the next section, as proof of concept, we discuss three projects by students at the Georgia Institute of Technology that were designed, tested with the plugin, and fabricated.

### *6.3.1 Rule 1: Vertical reinforcement*

One key factor in making composite masonry is to ensure vertical connectivity. Vertical reinforcement is the backbone of this type of structure. The fused rebar and grout serve as stripes of structural support that pass through the cores of the bricks. These bars must run continuously from the bottom to the top of the wall. This structural system is the main differentiator from traditional wet masonry, which relies on mortar in between brick bonds.

The vertical reinforcement rules check for two conditions, local and global vertical reinforcement. At the local level, each brick in the system must be connected to the main structure via at least one its cores. One of the cores in the brick must have enough overlap with the brick cores on the top and bottom that the rebar and grout can pass through the intersection and bond the bricks together (see Figure 6-3). This ensures that each brick is locked in place and supported by the overall structure. Once the local reinforcement condition is met, the structure must be checked for global vertical reinforcement. At the global level, a continuous stripe of reinforcement (i.e., rebar and grout) from the bottom to the top of the wall must pass through every brick in the wall. The overlapping cores create the negative space required for passing the rods and pouring the steel-fiber grout mixture (see Figure 6-3).

To check this rule, our algorithm creates the negative geometry of each cored brick (i.e., the geometry of the cores). For local reinforcement, we test the minimum overlap between the cores on consecutive rows. The overlap surface needs to have enough surface area to allow for rebar with the desired radius to be passed. For global reinforcement, all core solids in the wall are fused together (i.e., a Boolean union). Then, we check if the length of the vertical bonds created is equal to the length of the wall itself.

### *6.3.2 Rule 2: Transverse connectivity*

While Rule 1 checks the vertical structural integrity of a wall, the complimentary Rule 2 verifies the transverse connectivity of a structure. This is a condition required for prefabricated masonry so that structures can resist lateral forces during transportation, installation, and assembly as part of the final load-bearing structure. Transverse connectivity horizontally ties the vertically reinforced bonds to one another. There are two solutions to achieve this goal. The first provides connectivity by passing lateral forces through a connected chain of bricks. As illustrated in Figure 6-3, there should be at least two rows of bricks in which each brick is connected to two adjacent upper or lower bricks via two reinforced cores. This horizontal bond forms a zigzag chain that weaves the wall structure side to side. The second optional method enhances the structure by adding a reinforced bond beam at the top of the wall. In addition to transverse connectivity, this method can be used to attach lifting anchors and spread lifting loads across walls during transportation (see Figure 7-14b).

Here, our algorithm also uses bricks' negative geometry to check for Rule 2. For each two consecutive rows, the negative core geometries are created and the cores' solids on the top and bottom rows fused together. Then, the top row of bricks is moved up slightly so that the surfaces of the bricks on the top and bottom rows do not touch. In the next step, the three groups of solids (i.e., top row bricks, bottom row bricks, and fused core geometries) are merged. Once the result of the merge is only a single solid object, the wall pattern is considered to have passed the transverse connectivity requirement.

### *6.3.3 Rule 3: Corbelling control*

As opposed to Rules 1 and 2 that check for structural stability in the wall structure, Rule 3 verifies the constructability of the desired form. In this fabrication method, bricks are dry stacked (here, by a robot) before the steel rods are placed, and grout is poured into the brick cores to form the final composite. During dry stacking, each brick can corbel out from the wall, but some conditions must be met so that the unreinforced segment does not collapse. Corbeling control limits out-of-plane offset between a given brick and those that support it. First, each brick must be positioned in a way that the center of gravity of the brick is placed within the surface area of the brick below or on the surface on or between two bricks in the layer directly beneath (see Figure 6-3). Second, corbeling control needs to be performed on all of the rows of unsupported bricks prior to reinforcement. In this process, the robot stacks a few rows of bricks before the grouting process starts. Thus, while the projection of an individual brick may be acceptable, the placement of the brick may move the overall center of gravity of the dry stacked section such that it is out of support.

The developed algorithm holds the order of placement of each brick as a two-dimensional array. For each brick in the array, the algorithm calculates its center of gravity, checking the positions of the center overlaps on or between the surface areas of the bricks in the previous row. Next, it calculates the combined center for the brick and all bricks underneath it in that wall section (see Rule 4) and examines if it is supported by the rest of the wall assembly.

### *6.3.4 Rule 4: Structure segmentation*

Finally, Rule 4 ensures constructability of the structure by segmenting it in certain intervals. The overall wall structure must be broken down into every few groups of rows if steel rods are used for tensile reinforcement. As opposed to GFRP bars, steel rods have a high elastic modulus, meaning that they cannot be threaded between the sliding and shifting apertures in the cored bricks. Steel rods need to be cut into smaller lengths derived from the maximum curvature of the vertical cores and segment lengths such that passage of a straight 6mm steel rod is allowed. The rods require some overlap to ensure structural connectivity. The minimum requirement for the amount of overlap is equal to the height of one brick. This leads the minimum length of the rods to be equal to the height of three bricks (allowing for one brick to overlap on the top and bottom) (see Figure 6-3). Obviously, the maximum length of the rods cannot exceed the maximum height of the wall. A wall with less of a curvature would require less segmentation, which in turn would result in increased fabrication speed. The grout mixture takes about 24 h for the initial cure. As a result, the overall fabrication process takes N days, where N is equal to the number of segments.

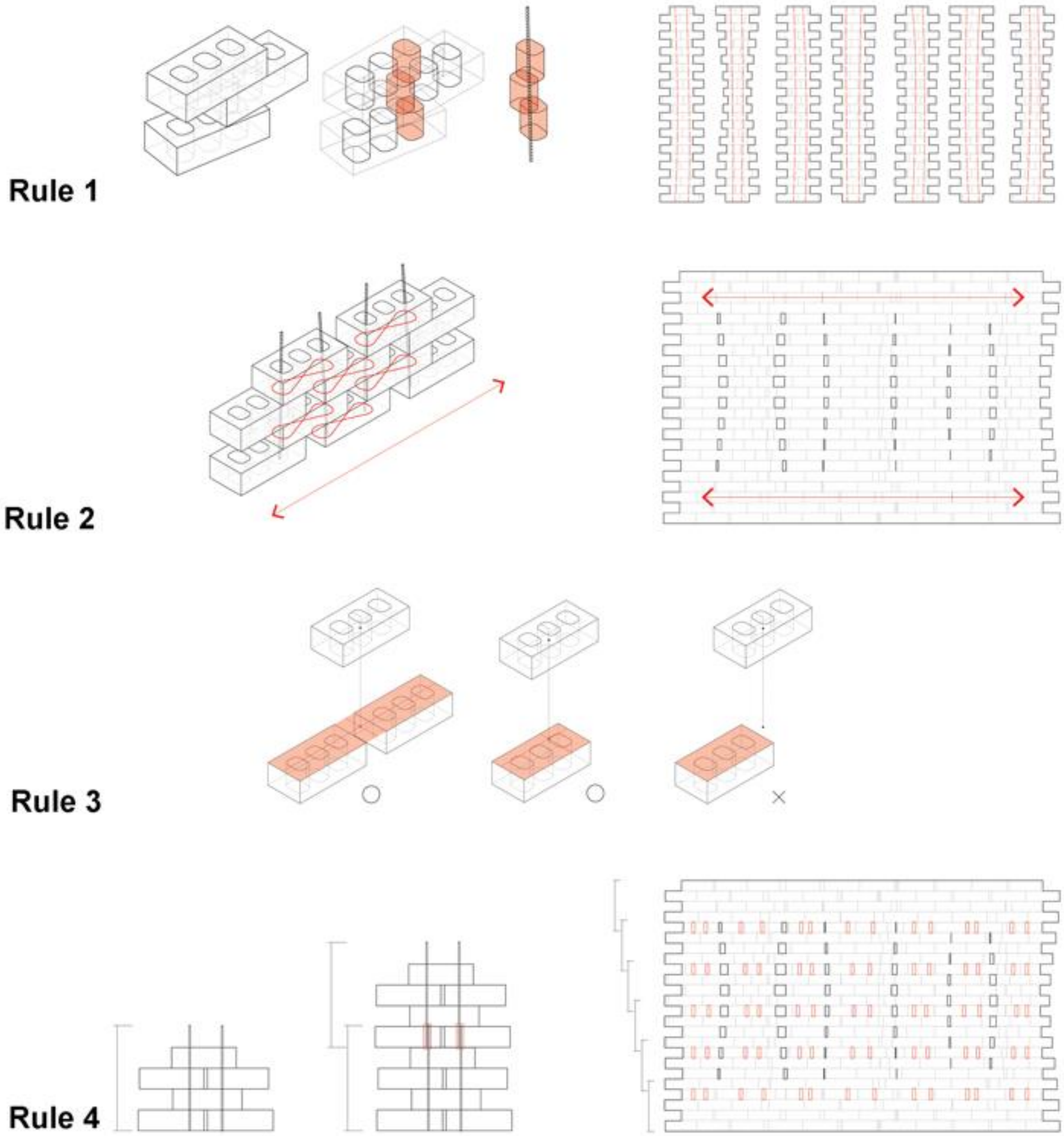


**Figure 6-3. Rule 1: Vertical reinforcement rule: local (left) and global (right) reinforcement. Rule 2: Transverse connectivity through a chain of cored bricks. Rule 3: Corbelling control. Rule 4: Wall segmentation to ensure constructability.**

### 6.4 Prefabricated Composite Masonry Experiments

In this section we present three experiments with composite masonry that were developed by architecture students at the Georgia Institute of Technology. One KUKA robotic arm, a KR Quantec Pro (KR 120 R2500) with a payload of 120 kg and arm reach of 120 cm, was used for these projects. The robot arm was equipped with a Schunk pneumatic gripper. Students designed and fabricated a material-handling conveyor for feeding brick units to the robot. Using this conveyor, the robot was able to pick up the bricks at a fixed position, simplifying the robot's motion programming.

**Figure 6-4. Project 1, wall with twisted pillars**

Students developed three algorithms to create various brick wall forms. Next, the patterns generated were tested for constructability by using our composite brick analysis plugin. The first project was composed of seven brick pillar forms that twisted around the z-axis and locked inside each other to form a wall pattern (see Figure 6-4).The next project was a double-curved surface where one central section pulled apart at the bottom to form two gaps in the wall structure (see Figure 6-5). The last project was a study on translating a grayscale pixelated image into brick rotation patterns (see Figure 6-6). Once the satisfactory forms were identified, students had to create the robot's operational program for the fabrication process. The KUKAprc plugin for Grasshopper was used to generate the robot’s toolpath and gripper operation for brick pickup and assembly.

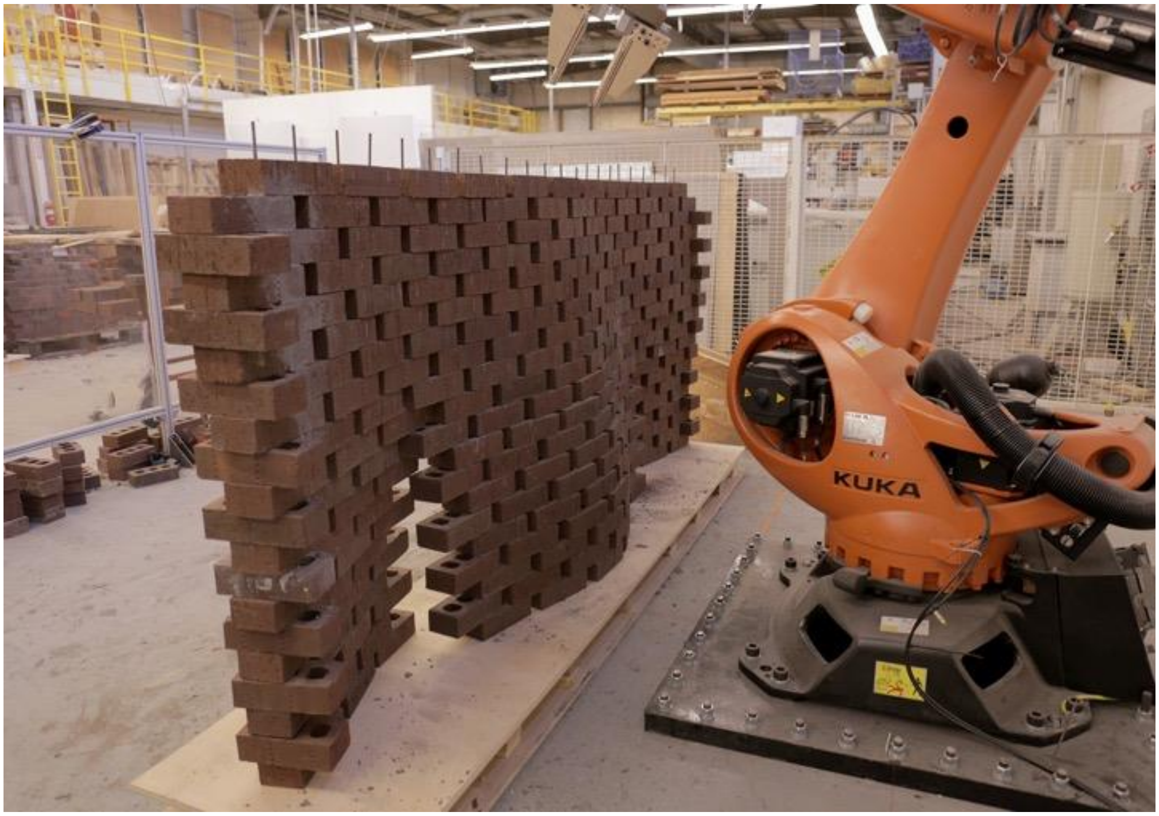


**Figure 6-5. Project 2, wall with a double-curved surface.**

**Figure 6-6. Project 3, wall with brick rotation pattern.**

For the construction process, we used 8 in x 4 in x 2 in three-core clay bricks, a self-consolidating grout mixed with steel fiber filaments, and 6 mm high-strength steel-threaded rods. The size and reach of our robot limited our wall structure's height to about 48 to 50 in. Based on the Rule 4 analysis of our structures, these 48 in walls were segmented into four brick rows, for a total of six segments per wall. Based on this information, we cut the steel rods into 10 in pieces. Our grout mixture needed approximately 24 h to cure for every segment of the wall before we could begin placing the bricks for the next segment. Each wall took about six days to fabricate, cure, and be ready for pickup with a forklift (see Figure 6-7).

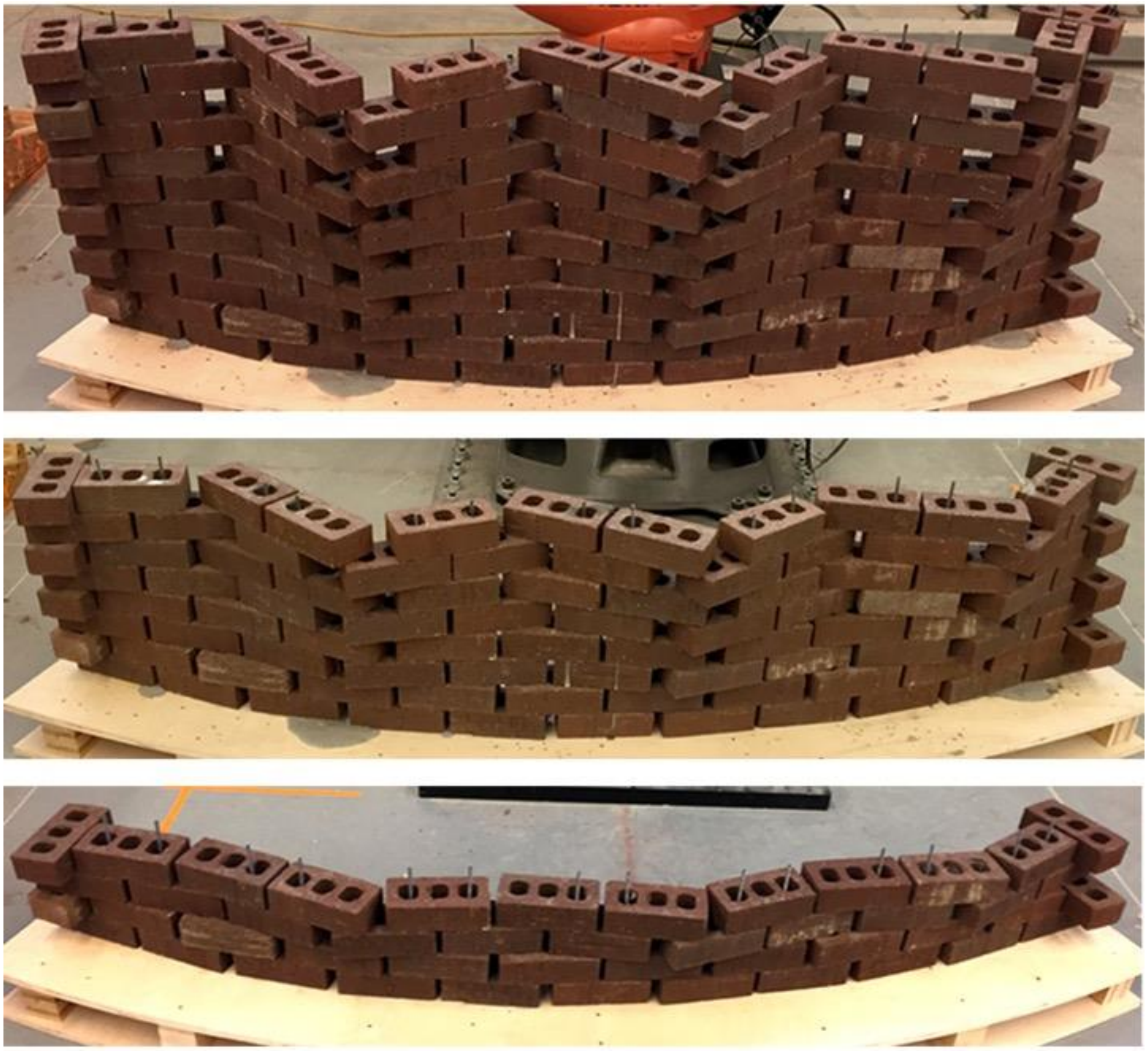

**Figure 6-7. Fabricating the wall in four-row segments.**

## 6.5 Conclusion

In this research, we presented a method for composite reinforced brick structures to be robotically prefabricated and transported to a construction site. The set of algorithms developed will help designers make more informed decisions in the early stages of the process. The three fabricated wall structures were moved successfully from the fabrication lab to the Georgia Tech campus, where they were displayed in an outdoor environment. As the next step of this research, additional exploration is required to connect the wall segments on the construction site to make a complete building structure.

# CHAPTER 7. ROBOTIC SHEET METAL FOLDING: TOOL VS. MATERIAL PROGRAMMING


**Abstract**

This research explores how deductive engineering thinking, as opposed to an abductive design rationale, can influence how robotic methods of fabricating building components are developed. The goal of this research is to demonstrate how creative thinking can introduce alternative robotic fabrication techniques targeted for the architectural mass-customization process. For this purpose, we chose robotic dieless sheet metal folding as the main fabrication technique, due to its wide range of applications in both the architectural construction and manufacturing industries. Two robotic sheet metal folding projects were developed. The first, an example of *tool programming,* took advantage of an engineering approach and was focused on the affordances of the tool (an industrial robotic arm). The second project, one of *material programming*, employed a design methodology and was directed towards the affordances of the material (i.e., stainless steel sheet metal). By discussing the advantages and disadvantages of each approach, this research argues that both engineering and design should be considered required and complementary processes in the development of new creative fabrication solutions, allowing them to and make the overall production process more efficient.


## 7.1 Introduction

A growing body of research is exploring novel methods for incorporating industrial robots in the digital fabrication and assembly of building components (Reinhardt, Saunders, and Burry 2016). Robots provide powerful and flexible fabrication tools. In the last two decades, the increasing affordability of industrial robots, along with the growing maturity of computational design software, has led to architects' adoption of these machines. By using industrial robots in the fabrication process, architects can eliminate the need for costly fixtures and setup times that hinder the cost-effectiveness of automation, especially with regards to physical prototypes with small lot sizes.

However, the robot control and motion programming tools that are currently being adopted by designers were all originally developed for engineering-based manufacturing industries (Kolarevic 2004). This technology facilitates the industrial mass production of components with known properties and processes, and thus predictable outcomes. The current industrial robot control systems require that the designer comprehensively understand the design object and embed detailed design and machining data in the digital model before the fabrication process begins. Consequently, the process of design to fabrication is mostly a one-directional workflow in which the designer must predict the material state, tool selection, fixture positioning, and robot motion planning, usually based on prior experience.

Focusing on this current technological gap, this research explores how engineering versus design methodologies might influence a designer's approach to selecting and applying a fabrication method and process. In other words, this study demonstrates how design thinking can put forward alternative uses for robotic technology in the fabrication process. For this purpose, two robotic fabrication projects were completed, and then examined, and compared. The design intent for both was to develop an aesthetically and functionally effective shading canopy to be fabricated and installed on a building's balcony. Both projects were composed of mass-customized and geometrically parametric tessellated structures, which were fabricated using robotic fabrication techniques and assembled manually.

The first project focused on the affordances of the tool (an industrial robotic arm) or *tool programming* (see Figure 7-1, left). By tool programming we mean an engineering approach to design and fabrication, where the desired geometry is established *a priori* and the materials, tool paths, and fixtures are selected to generate that desired geometry within the constraints of the system. The second project addressed the affordances of the material employed, or *material programming* (see Figure 7-1, right). By material programming we mean a design approach in which the concept is developed as a result of experimentation focused on the reciprocal exploration of form, testing of the material's behavior, and robot kinematics.

Many of the objectives of design-based fabrication are similar to those of the engineering-directed approach, such as fewer operations, reduced operational complexity, ease of reconfiguration, decreased lead time, minimization of material waste, and less rework (Qattawi et al. 2014). The main difference, however, is that engineering processes (i.e., *tool programming*) are based on deductive reasoning, increasing the functionality of the operation and thus facilitating standardization for the mass production of parts. Conversely, design-based processes (i.e., *material programming*) relies on abduction (Kolko 2010) and prioritizes customization of parts production, with an emphasis on creativity and uniqueness.

**Figure 7-1. Left: Project 1, programming the tool; Right: Project 2, programming the material.**

## 7.2 Robotic Dieless Sheet Metal Fabrication

Robotic sheet metal folding was chosen as the main material and fabrication technique for the case studies examined in the present research. Sheet metal forming is one of the oldest and most well-studied metal fabrication processes (Duflou, Váncza, and Aerens 2005). Sheet metal has a wide range of applications in both architecture and manufacturing. It depends on deformative fabrication techniques by which three-dimensional (3D) forms can be achieved from two-dimensional (2D) planar materials. As opposed to subtractive manufacturing methods such as milling, deformative fabrication provides considerably less material waste. In addition, sheet metal (usually steel, stainless steel or aluminum) offers greater structural stiffness and strength compared to unreinforced plastic or the powder-based materials currently used in robotic additive and 3D printing processes.

The main advantages of robotic dieless sheet metal fabrication over other numerically controlled machines (e.g., sheet metal CNC press brakes, shears, and punching machines) are the flexibility and adaptability of robotic arms for performing different fabrication and assembly tasks with appropriate end-effectors. Robotic dieless sheet metal fabrication eliminates the need for more time consuming and expensive methods such as drawing, punching, and hydro-forming with dies (Liao and Wang 2003). Utilizing robotic arms eliminates the need for the production of dies, molds, and presses. In addition, robotic arms can produce extremely complex geometries that otherwise could be achieved only through traditional manual metal craftwork, a slow and time-consuming process. Robotic dieless sheet metal fabrication is most suitable for making test models with short setup times, prototypes with many parts of unique geometric shapes, and products in small batches for which the cost of making dies would be very high.

### *7.2.1 Fabrication Process*

Currently, there are three main dieless techniques suitable for robotic sheet metal fabrication. These include: (1) folding or bending (e.g., straight-line folding for simpler geometries (Lavallee, Vroman, and Keshet 2011, Liao and Wang 2003, Aomura and Koguchi 2002) and curved folding for more complex 3D geometries (Epps and Verma 2013)), (2) incremental forming (Kalo and Newsum 2014, Ponticel 2013, Mohanty, Regalla, and Rao 2015), and (3) metal driving (i.e., shrinking and stretching) (Opritescu and Volk 2015, Hoffmann, Hautmann, and Petry 2005, Nicholas and Rossi 2018). This research focuses on the straight-line folding technique because it can be considered the most basic variant, offering less complexity and fewer factors affecting the design and fabrication process. Parts with complex geometries can be folded from a single planar sheet of metal, without stretching or cutting (Tachi and Epps 2011, Epps and Verma 2013, Kilian et al. 2008). Sheet metal folding changes a part's geometry by adding a V-shaped section along a straight or curved axis. This process is based on the plastic deformation of the sheet metal and employs the material’s ductility. By using this technique, we can achieve the desired geometric shapes while also increasing the strength and stiffness of the resulting part.

In the manufacturing industry, the terms “folding” and “bending” are often used interchangeably. However, in this research, we distinguish between the two. By bending, we mean processes that use various punch die setups or roll forming. In addition to the V-shape, bending can produce various figures based on the shape of the die, including U and channel shapes. Bending flat sheets along straight lines in press brakes is the most common method. Boxes, brackets, and similar shapes are possible by repositioning the workpiece in the brake, with subsequent bends made perpendicular or oriented at other angles relative to the prior bends. However, when using press brakes and punch dies, only certain shapes and angles are feasible. This is due to interference between bent shapes on the workpiece, or the workpiece and press brake. Thus, human skill or advanced planning algorithms are required in the design and selection of the bend sequence (Duflou, Váncza, and Aerens 2005). Collision detection algorithms based on segment intersection have been developed to create piecewise linear curves with a series of straight-line bends, with the bends all parallel to one another. With advanced planning, it is possible to check for interference (Liao and Wang 2003).

By folding, we refer to the dieless sheet metal forming process. Folding is achieved by applied directional force via a set of grippers along a weakened axis line; this weakening can be realized by techniques such as grooving, perforating, or notching. While bending can be used for thicker materials to produce various bent shapes and radiuses, folding can potentially eliminate the need for dies and brakes, providing flexibility in terms of the geometric forms generated. Bending requires specific tooling for each desired shape and angle of bend, which makes it more suitable for repetitive processes where thousands of similar objects must be made. Conversely, folding is a more flexible process appropriate for unique geometries and small numbers of parts, such as in architectural design projects.

In order to take the most advantage of the range of movements made possible by robotic arms and achieve a wider range of possible angles, we adopted a technique in which folds are made by applying directional force to a weakened material on a bend line. Compared to methods that use a press brake for folding, this technique allows for fewer robots and supporting tools (i.e., one robot with a gripper arm and stationary gripper). Below, the fabrication process, required tools, material settings, and geometric model requirements for dieless folding are discussed. Based on these factors, two case studies are then examined. The first focuses on a fabrication process based on the constraints of a tool system, and the second considers the correlation between sheet metal behavior and various design and fabrication decisions.

### *7.2.2 Parts Preparation*

The process for folding sheet metal consists of several steps, including cutting, perforating, and folding (see Figure 7-2).

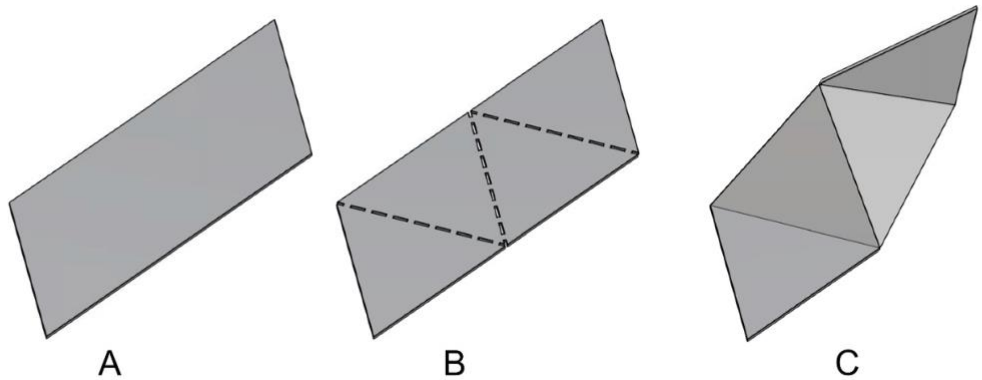


**Figure 7-2. Sheet metal folding process: (A) cutting, (B) perforating, and (C) bending.**

In the first step, the workpiece must be cut out of a manufactured metal sheet, which comes in a variety of standardized sizes. For this purpose, the unfolded or unrolled two-dimensional geometry of the designed part is needed. Development of the desired flat layout for the intended structure requires that the 3D geometry of the part be transformed into a 2D layout. In that way, it can be formed out of the flat sheet metal. Flat layouts are generated via a process of unfolding or unrolling, where the only condition is that the flat pattern generated is not self-intersecting. While many CAD tools such as Rhinoceros 3D and SolidWorks have algorithms for unfolding or unrolling 3D objects, in the general case there is no evaluation process for self-intersecting faces or means of finding the most suitable flat pattern for a specific folding process, based on the selected tools and materials.

Nevertheless, the generation and evaluation of a flat pattern can be performed both manually and computationally. Computer geometry techniques take advantage of the topological data of the object as defined by the connectivity between bending edges and faces. This topological data is represented in the form a graph called a connectivity graph (Aomura and Koguchi 2002) or face adjacency graph (Liu and Tai 2007, Qattawi, Mayyas, and Omar 2013). Connectivity graphs are undirected graphs in which each node represents a face, and the link between two nodes indicates the edge between two faces (see Figure 7-3, left). This type of graph can be used as base data for unfolding and evaluation algorithms. Generating all of the spanning trees for a connectivity graph produces all of the possible flat patterns of the associated folded 3D geometry. Some of the possible spanning trees and their corresponding flat patterns for a sample 3D object are illustrated in Figure 7-3. Different search algorithms, such as depth first search (DFS), breadth first search (BFS), A*, and genetic algorithms can be applied to face adjacency graphs to create unfolded flat patterns for an intended object.

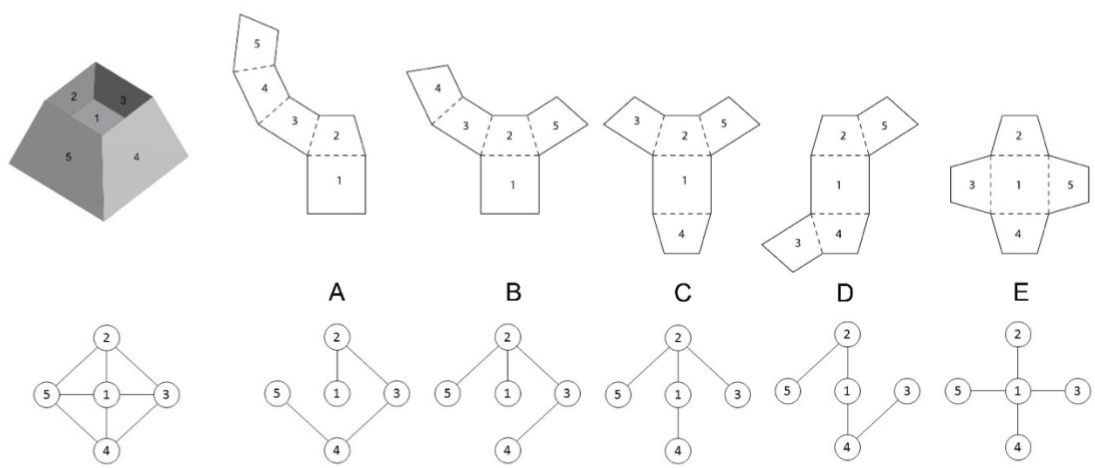


**Figure 7-3.Representation of a sample object's topology with a face adjacency graph (connectivity graph).**

Since the process of folding, unlike bending, relies on a dieless solution, it is necessary to enforce the fold along the desired axis line. Folding techniques for metal are different from those for other materials such as paper, due to the material’s thickness and rigidity. The main solution to this problem is the localized reduction of stiffness along fold lines, which can be accomplished via techniques such as groove (or scored) joints, or more commonly, reduced-area joints with perforation (see Figure 7-4) (Delimont, Magleby, and Howell 2015, Chen, Peng, and You 2015). For grooving, the geometry consists of a line placed on the intended bend axis. The line's thickness determines the depth of the cut, which should be less than the material's thickness (see Figure 7-4A). Perforation geometry is a series of line segments and the spaces between them, where each line segment represents a complete cut through the material (see Figure 7-4B).

Perforation locally weakens the material in flexure, allowing it to be bent at the location of the perforation and ensuring that the dieless bend occurs at the precise location. However, it must be acknowledged that perforation permanently weakens the part in flexure. Therefore, final geometries that include the overall triangulation or trussing of the assembly are likely to be more successful, as these forms transmit forces more as in-plane tension and compression, and less as bending. The ability to carry localized bending forces between perforations is therefore preserved.

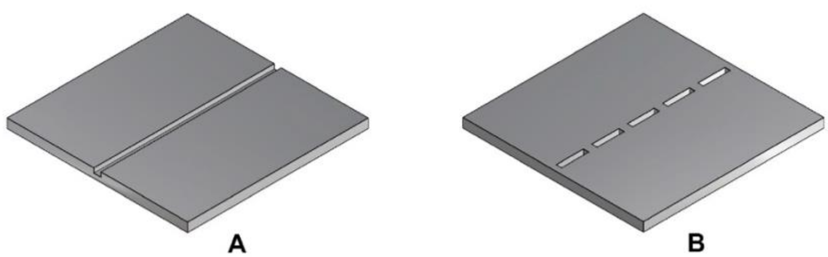


**Figure 7-4. Samples of sheet metal surface reduction axis lines: (A) groove joint and (B) perforation.**

While shapes with simpler geometries can be cut with tools such as sheet metal shears (either manual or CNC), more complex shapes require implements capable of handling more complex 2D geometries such as waterjet cutters, laser cutters, plasma cutters, and CNC milling machines. Some are inherently suitable for cutting sheet metal (e.g., waterjet or plasma cutters), while others such as laser cutters should have specific characteristics to accommodate this type of material's reflectivity and thermal conductivity. Moreover, most industrial lasers cannot cut through thicker sheets. In the present research, a three-axis OMAX 60120 waterjet was used for cutting and perforating processes.

### *7.2.3 Material Considerations*

Folding sheet metal produces shapes via the exertion of bending stresses that exceed the material's yield point but that are below its maximum tensile strength. Springback is the tendency of the bent material to partially return to its original shape once it has been released from the forces created by the forming tool (Benson 2014). When folding, sheet metal should be overbent beyond the desired angle to compensate for springback. Consequently, prediction of the final geometry after springback is critical. The main determinants are the tensile strength and thickness of the material, type of tooling, and characteristics of the bending mechanism. Perforation is an important step in reducing springback in dieless sheet metal folding.

In this research, 20-gauge stainless steel sheets were used. We chose this material because it is corrosion resistant, so the final parts would not need additional rust prevention and surface treatments such as paint and other coatings. Thus, it would be a suitably durable material for a shading structure. In order to assess the required material properties, we devised a project-specific test to measure the springback of these steel sheets.

For this test, two sets of 4" x 8" pieces of metal were cut with a perforated line at the center of each rectangular piece (see Figure 7-5). Each set included 12 cut and perforated parts. In order to measure the amount of springback relative to the bend radius, we bent each piece manually from 20 to 130 degrees in increments of 10 degrees. After each bend, the final bent angle of the piece (which was the result of springback after the bending process) was measured. In addition, we explored the influence of different perforation patterns of the bent angle on the given material, after springback (see Figure 7-6). Type A used a perforation pattern of ¼” void and ¼” solid, whereas perforation Type B used a pattern with ½” void and ½” solid. Figure 7-5 shows the results for the two different perforation types and their influence on the bend angles and final angles.

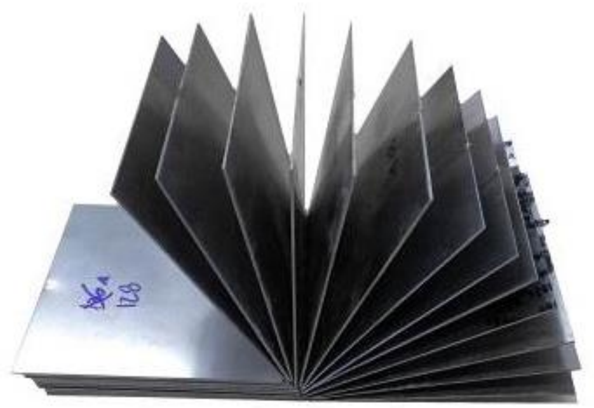

**Figure 7-5. Set A of the parts bent from 20 to 130 degrees in increments of 10 degrees.**

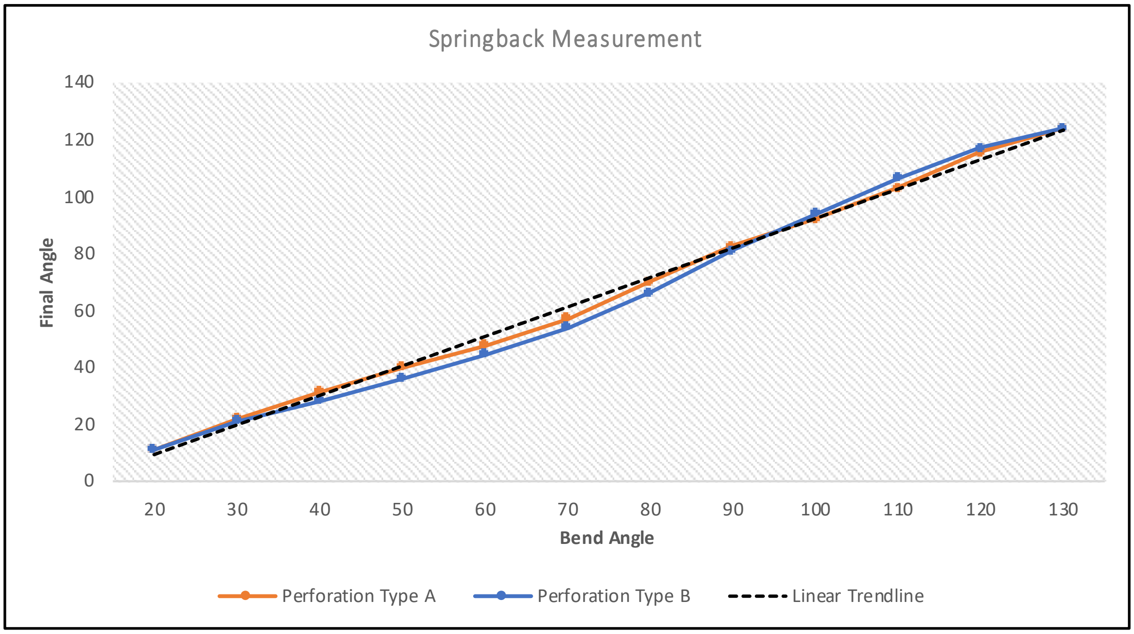


**Figure 7-6. Springback measurements for two sets of bends with different perforation patterns.**

Based on the results of this test, the perforation pattern was not found to be significantly impacted by the amount of springback. Sets A and B both showed very similar final results in terms of bent angles. Springback had a linear impact on the final bent angle within the range of angles tested for this experiment. The mean of the angle difference between the bent and final angles was 8.7 degrees for Set A and 9.75 degrees for Set B. For this research, we rounded the adjusted angle to 9 degrees. This meant that, for example, in order to achieve a final bent angle of 45 degrees, the robot needed to bend the part 54 degrees to compensate for springback.

### *7.2.4 Fold planning*

Folding requires planning to determine a feasible optimal sequence of folding operations (Jiménez 2012). For workpieces with more than one fold, interpreting the part's geometry and planning the bending sequences are highly important tasks in the process

(Aomura and Koguchi 2002). Dieless folding is performed with a robotic arm equipped with a gripper and at least one other supporting gripper. The secondary gripper can either be stationary or mounted on an assistant robotic arm. Fold planning includes steps for positioning the bend lines relative to one another and the overall geometry, a combination of multiple bend angles and their respective impacts on the fold sequence, setback calculations (i.e., the distance between gripper sets and the fold line), and collision detection and avoidance.

Each fold sequence plan requires input and output data. Input data for a fold planner (either manual or computational) includes the flat cut and perforated parts, final 3D form, tools and settings, and finally the work environment and all relevant obstacles. The input data are supplemented with feasibility constraints such as material specifications, collision avoidance, and tolerance specifications. The output involves the setup and instructions; in the case of robotic folding, this would involve the robot's tool path and supporting open and closed sequence instructions for the grippers.

The steps followed in the output folding sequence are: (1) placing a flat workpiece on the holding/feeding platform, up against a back gauge so that the part is precisely located; (2) finding the best grasping position for each bend; (3) determining the best robot toolpath for moving the part to the folding position and avoiding collision; (4) setting up the robot's movement speed to avoid part vibration; (5) bending the part, after consideration of the springback factor; (6) finding the required repositioning and regrasping moves; (7) sequencing opening and closing of the robot and assistant grippers; and (8) offloading the folded part. The general steps for a fold sequence are illustrated in Figure 7-8.

This process can be straightforward if the workpiece only requires a few folds. However, sheet metal fold sequence calculation and evaluation is a combinatorial problem. When the number of folding steps increases, the possible number of bending sequence combinations scales up rapidly. Theoretically, for a workpiece with n number of folds, there are $2^n n!$ different fold sequences (Duflou et al. 1999). In addition, a validation process is required to determine the feasibility of the computed fold sequences. Unworkable sequences may include obstructing folds on the workpiece itself, or collision of the workpiece with the robot and/or supporting tools. For complex forms, interpretation of the part's geometry and planning the bending sequences require either the expertise of a human process designer or advanced computer optimization methods (Duflou et al. 1999, Aomura and Koguchi 2002, Kannan and Shunmugam 2008).

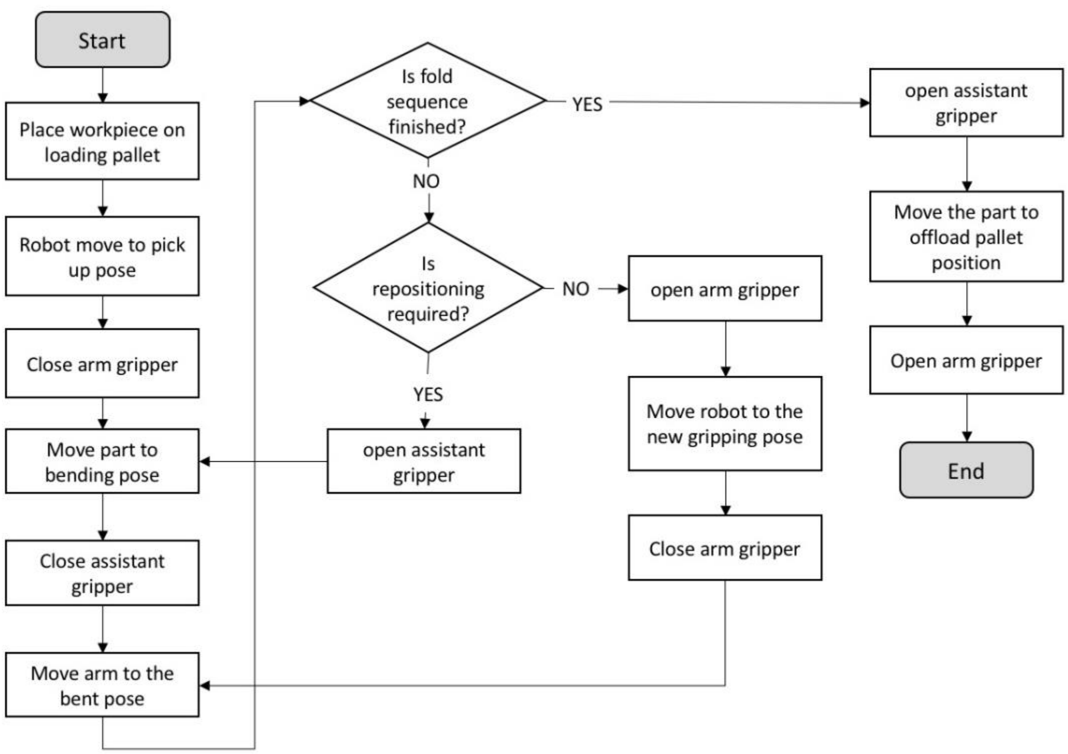


**Figure 7-7. Robotic sheet metal folding process.**

## 7.3 Robotic Folding Projects

We completed two different folding projects to study the effects of the different design and fabrication factors discussed in Section 7.2. The main goal motivating these projects was the development of a non-load bearing hanging outdoor canopy. Although, the projects had different geometric designs, they both used the same material, tools, and environment setup for fabrication. The design intent for these projects, and thus their geometric designs, dictated two approaches to the folding process. The first focused on *tool programming.* A 3D geometric form was designed in advance, and the fold sequence and robot tool path were generated to realize that exact geometric form. The second project addressed *material programming*. This was a reciprocal exploration of form generation, material properties, and robot movements, the goal of which was to reduce the need for meticulous fold sequence planning.

One Kuka robotic arm, a KR Quantec Pro (KR 120 R2500) with a payload of 120 kg and arm reach of 120 cm, was used for both projects. The robot arm was equipped with a Schunk pneumatic gripper. As the sheet metal folding process requires at least two grip points, we designed and fabricated a fixed grip system with two double-acting air-powered vises connected by two 12 mm steel plates (see Figure 7-8). These pneumatic vises were linked to the Kuka robot's digital outputs so that they could be controlled via the Kuka robot language (KRL) code. This setup limited the part size to an approximate maximum of 100 cm in length. To generate the robot's toolpath for the folding sequences, we used the Kuka|prc plugin  for Grasshopper 3D, a graphical algorithm editor . The digital outputs for the control of both the Schunk gripper and pneumatic vises were also programmed by Kuka|prc, which was embedded in the final generated KRL code for controlling the robot.

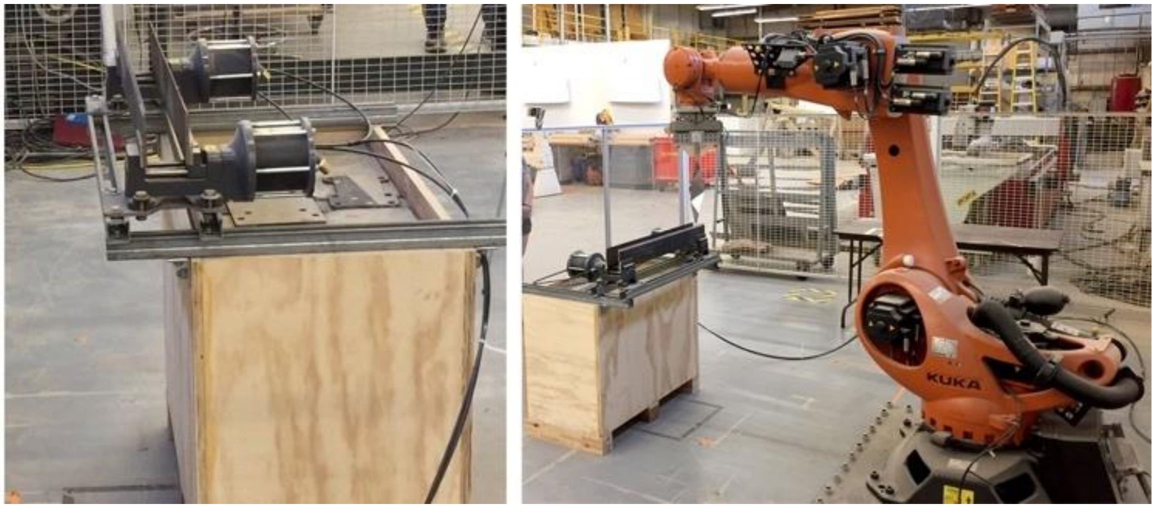


**Figure 7-8. Tools and environment setup.**

### *7.3.1 Project 1: Programming the Tool*

This project relied on an engineering approach to programming the tool used to fabricate the desired geometry and achieve the project’s goals. Students designed a tessellated structure composed of folded triangular modules. The final design was guided by the goal of creating complex configurations and effects while also reducing the number of folds and simplifying the fold sequence. The modules developed were based on an isosceles right triangle, which was subdivided into two triangles and a quad with four straight-line folds. The internal subdivision lines were used as fold lines at 135 and 225 degrees (see Figure 7-9).

**Figure 7-9. Project 1 module, fold lines, and fold pattern.**

The modules were then divided into two groups based on the direction of their folds: Group A's fold direction was the mirror image of that of Group B. Groups A and B were placed next to each other in the final assembly. The mirrored geometry of the modules provided the geometrical connecting points, as well as the negative open spaces in the assembled structure that were necessary to create shade and light patterns desired for the canopy (see Figure 7-10, left). The individual modules were connected with a square flat plate at each corner (see Figure 7-10, right).

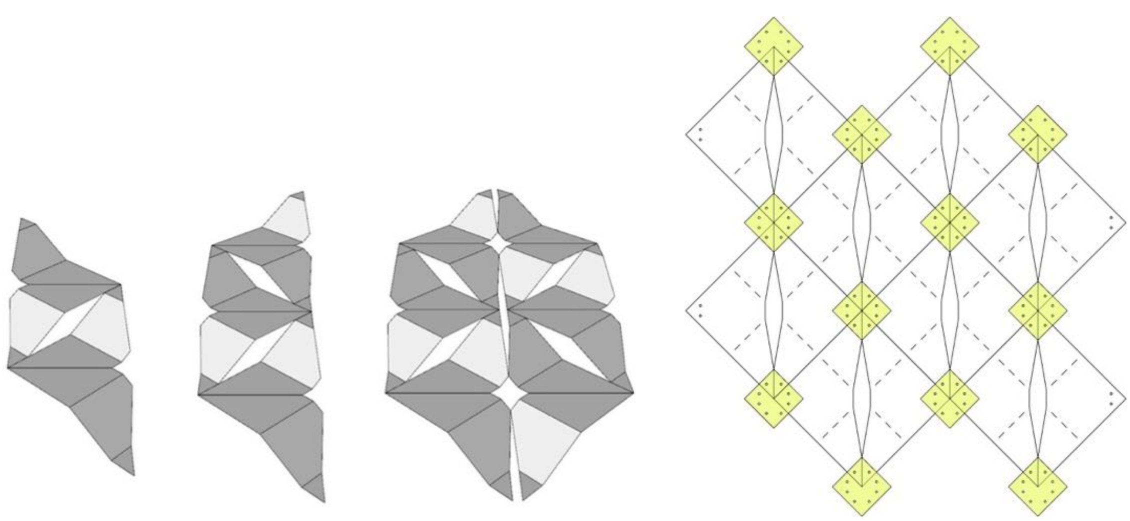

**Figure 7-10. Assembly sequence of the A and B modules.**

As discussed in Section 7.2.3, springback had to be calculated and incorporated into the folding process for each fold line in the module. Based on the results of the springback study, no significant difference was observed between the two perforation patterns on our selected material with the intended scale and dimensions. Consequently, the perforation pattern with longer cut lines and gaps was selected for the final stage. This perforation pattern would save water jet machining time, a significant impact factor when cutting a large number of parts.

The results of the springback study showed that the difference between the folded and final angles was an average of 9 degrees. This offset was incorporated into our robot toolpath model in the generative script. The Grasshopper script was modified to automatically add an additional 9 degrees to all of the robot's rotational movement in the fold steps. There were multiple fold sequence options for any part with more than one fold line, as discussed in Section 7.2.2. In this project, there were four fold lines on each module, resulting in 4! = 24 different possible fold sequences. However, based on the tool capabilities, dimensions, and setup for this project, the fold sequence illustrated in Figure 7-11 was selected for the fabrication process.

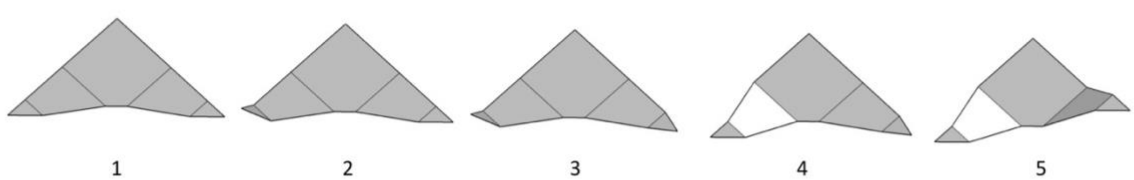


**Figure 7-11. Fold sequence for the first project module.**

The folding process began with folding one of the smaller triangle-shaped flaps at one end of the module. By starting with the smaller flaps, the larger section could be held by vice grips to provide the greatest amount of stability during the folding process, reducing deformation. To fold the second small flap on the opposite side of the module, the module needed to be lifted by the robot arm, rotated 180 degrees around the Z-axis, and placed back into the vice grips. This rearrangement was required to accommodate the robot's possible joint configurations and avoid collisions between the part, robot arm, and vice grips. After folding the second small flap, the larger sections on the module needed to be folded. After the first large fold, the module again needed to be rotated 180 degrees before the robot could perform the process to create the second large fold. After this step, the part was moved back to the feed table, marking the end of the process (see Figure 7-12). Performing each fold increased the complexity of the part's geometry, gripping location, and folding sequence. While the first fold was applied to a flat geometry, subsequent folds had to consider a complex 3D folded component.

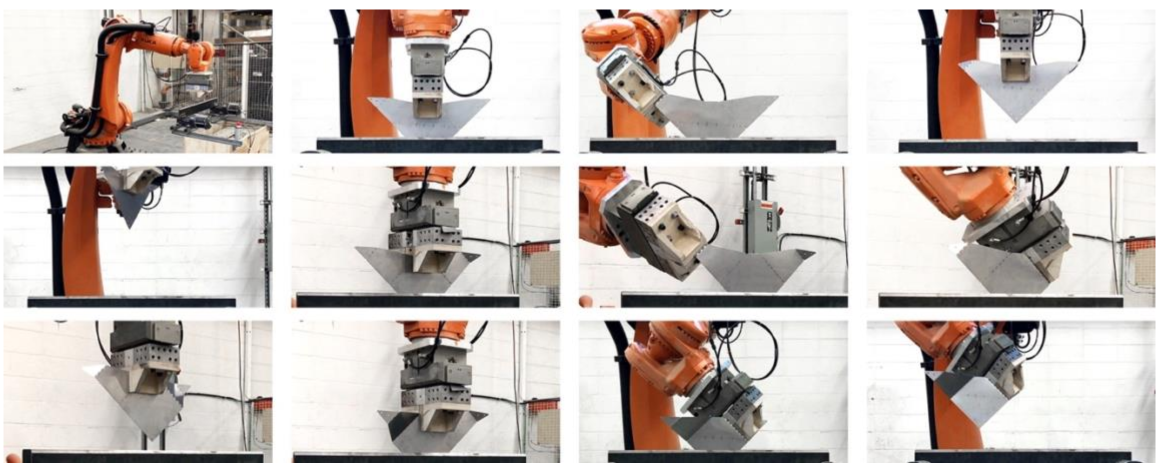

**Figure 7-12. Project 1: robotic bending sequence.**

Two sets of modules with mirror geometry were fabricated for this project. They were connected via square-shaped plates that were riveted to the modules. The assembly process was performed manually (see Figure 7-13).

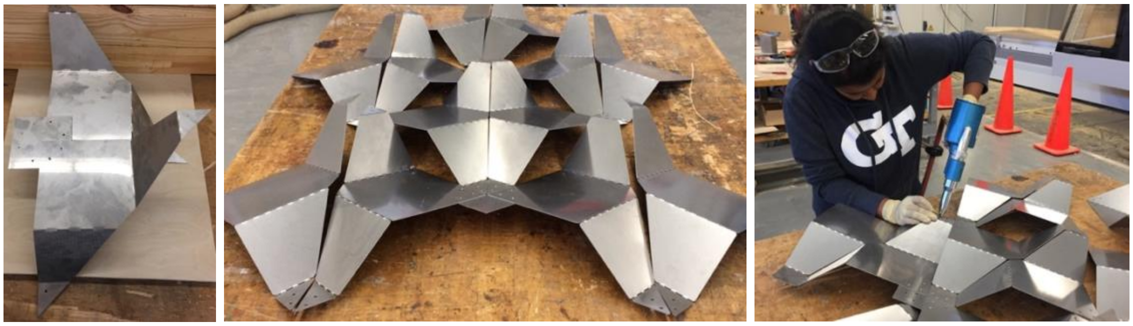

**Figure 7-13. Assembly process.**

The strength of the project relied on deductive reasoning to increase the functionality of the operation, a methodical approach to the fabrication process. The whole setup was intended to increase functionality in the operation. Once the logic was set and the toolpath generated, the result was predictable, reliable, and transferable to other projects with similar fold geometries. The elements of the process are parametric variables comprised of the stances of the general algorithm shown in Figure 7-7.

Conversely, programming for the robot toolpath was laborious and complex. This workflow was composed of iterative sequences of robotic movements, repeated opening and closing of the robot arm's gripper and vice grips, part grasping, repositioning, and bending. For each bending step, the springback factor, best grasping position, and collision avoidance all had to be calculated and incorporated into the robot's toolpath. The final result was a relatively sizeable set of code for which error detection and debugging or changing any other parts of the program were both complicated and time-consuming. For mass-customized products, the longer development and setup time could be compensated for by larger production volumes. However, for a low-volume mass-customized production process, faster development and setup stages would be desirable, since the need for change and adaptation to updated design criteria, new parts, and additional projects may be frequent. The second project for this study focused on reducing the lead time by challenging this design-to-fabrication methodology.

**Figure 7-14. Project 1: the final prototype.**

*7.3.2 Project 2: Programming the Material*

Learning from and building upon our first project, in our second we focused on a design approach to the fabrication process. For this endeavor, we took advantage of the inherent properties of the sheet metal in an effort to drive the fabrication method and achieve the design intent for the final product. Recent design research has considered new design and fabrication projects that take advantage of construction materials' behaviors and smart interventions in assembly systems. A material's properties and behavior in different environments and in response to various construction techniques can be used to derive factors affecting design generation, leading to new fabrication methods (Menges 2012a). There is a growing body of research on material programming and how the type and behavior of the material, as well as its function and fabrication requirements, can impact the fabrication process.

One group of studies focused on enhancing or altering the behaviors of flexible materials such as wood via interventions like partial material removal and/or weakening. Incisions made in different directions (e.g., along or against the grain, on one or both sides of a sheet, etc.) and calculated material removal by CNC milling or laser cutting can make the material more flexible in certain directions. Examples of these programmed materials exhibiting two or three-directional flexibility can be seen in products developed by Dukta Flexible Wood (dukta-gmbh) and kerf-based complex wood systems produced as part of the Performative Wood Studio at the Harvard Graduate School of Design (Menges 2012b). A second body of research took advantage of properties inherent in the material, using them as driving design factors; examples include the humidity responsiveness of veneer-composite elements (Menges and Reichert 2012). Other work has sought to develop new environmentally-responsive materials that would allow for the direct control and programmability of physical material transformations (Correa et al. 2015, Tibbits 2014, Oxman 2012).

Accordingly, in this second project, we investigated a technique that would take advantage of the inherent ductility of sheet metal in order to develop a method for reducing the effort required for fold sequence planning. The main goal was to design a folding technique for metal that could achieve multiple folds on one module with as few robot movements as possible, using advanced planning of the crease lines and tool movement in accordance with the final shape. This concept originated from a simple twisting of a strip of material. It relied on the fact that material reduction on crease lines creates areas with lower yield points than the rest of the material. The hypothesis was that multiple folds on a module can be achieved through a twisting motion (see Figure 7-15). Unlike in Project

1, the force direction was not perpendicular to individual fold lines, but rather followed the transition path between the 2D geometric state and final 3D state of the module.

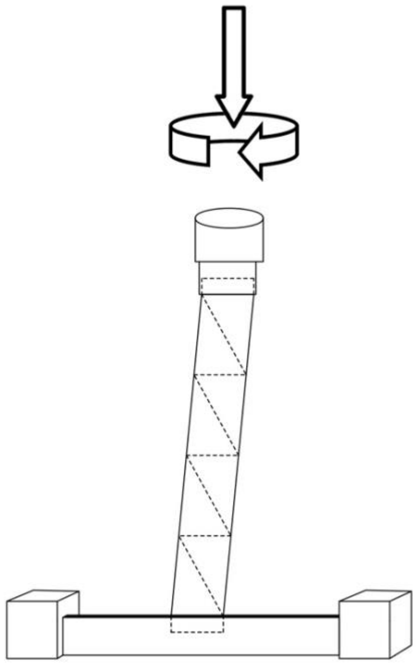

**Figure 7-15. Concept design for the continuous folding of sheet metal.**

In order to test the hypothesis, we needed to find the correlation between the placement of crease lines derived from the final desired geometry and the precise identification and application of force direction. We referred to this correlation as the "programming" for the material. The main challenge was the prediction and calculation of factors affecting the outcome of the fabrication process. These factors included the position of bend lines, combination of multiple and various bend angles, effect of springback on the overall process based on the material's thickness and type (e.g., mild steel, stainless steel, aluminum), perforation geometry, and setbacks (i.e., the distance between each gripper and fold line). Achieving the optimum programming for the material relied on numerous physical tests, accompanied by analytical models such as FMA. Physical testing involved the reciprocal study of the effects of the design, material, tools, and fabrication process on one another and the final outcome.

Consequently, we designed a series of tests for this process. First, we focused on exploring the relationship between the fold pattern (i.e., placement and number of crease lines) and final 3D geometry achieved after the twisting operation. Initially, the flat cut parts were twisted manually using two grippers, but the action applied the logic of the robotic folding process. (see Figure 2-9, left). Once the most suitable relationship between the final geometry and fold pattern was identified and the fabrication logic of the part established, the results of the manual study were repeated using the Kuka robot and stationary gripping station (see Figure 2-9, right). A second series of tests were required in order to find the best robot toolpath for the fabrication operation. The robot's toolpath was generated and simulated with the Kuka|prc plugin for Grasshopper, and then tested physically on flat cut materials.

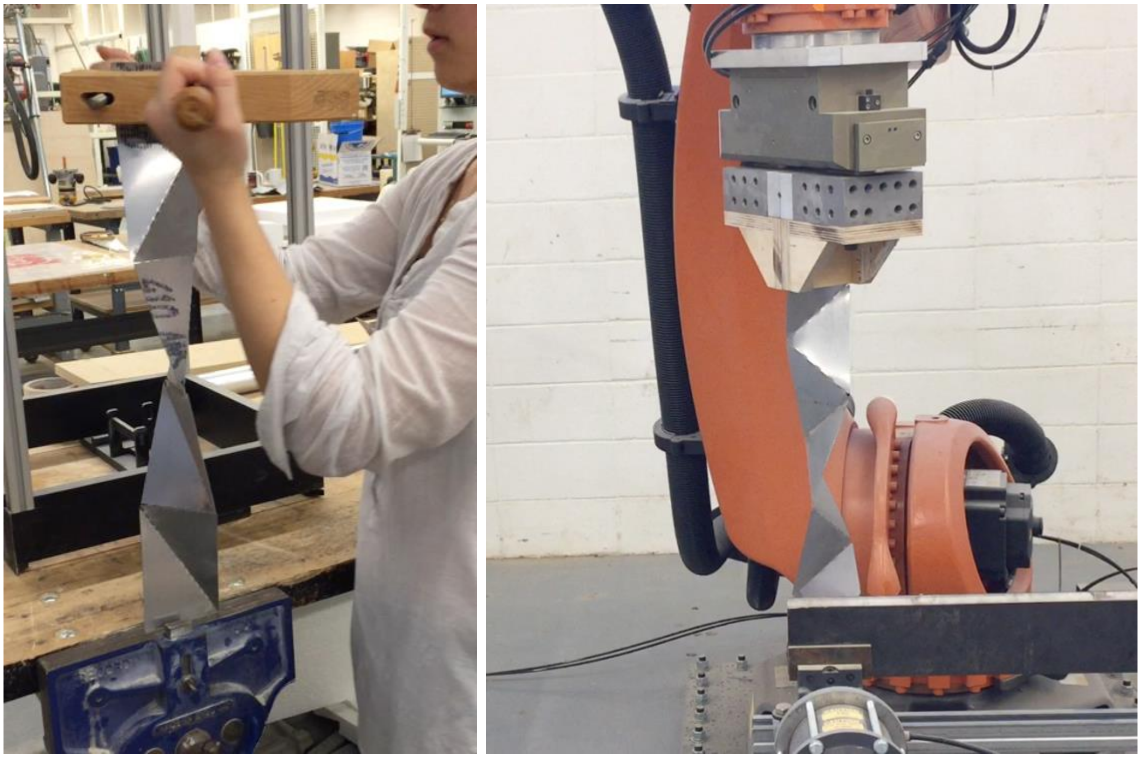

**Figure 7-16. Manual folding test vs. robotic folding test.**

In order to examine the relationship between the fold pattern and geometry of the folded module, a few parameters had to be kept constant, including the length of the metal strip, final twist angle after springback (i.e., 180 degrees), perforation geometry, and material type and thickness. The variable factors were the fold pattern and angles of fold lines relative to the twist direction. For the fold pattern, three main sets of geometries were tested: parallel fold lines oblique to the strip's main axis, forming parallelogram-shaped subdivisions; fold lines as continues zigzag polylines, making right triangles; and fold lines as continues zigzag polylines, creating isosceles triangles (see Figure 7-17).

The parallel fold lines proved to be unsuitable for our design intent. This fold pattern created an almost tubular folded part; also, the fabrication process could not be achieved with a single twisting action. Next, different variations of the zigzag fold lines were tested for both the fabrication process and final geometry. The factors tested were the number of fold lines, angle between the fold lines, and angle of the fold lines relative to the strip. Tests of the right triangle zigzag pattern showed that only a few of the metal strips folded only along certain fold lines during the folding process. Other fold lines remained almost in their original state. The result was an unevenly folded strip that in most test cases bent only at the center of the part. For this fold pattern, the number of subdivisions and amount of rotation had no significant impact on the results. The third set of patterns, the isosceles triangle-forming zigzag fold lines, demonstrated a gradual transformation along the entire fold segment and produced outcomes that were very close to the initial design intent.

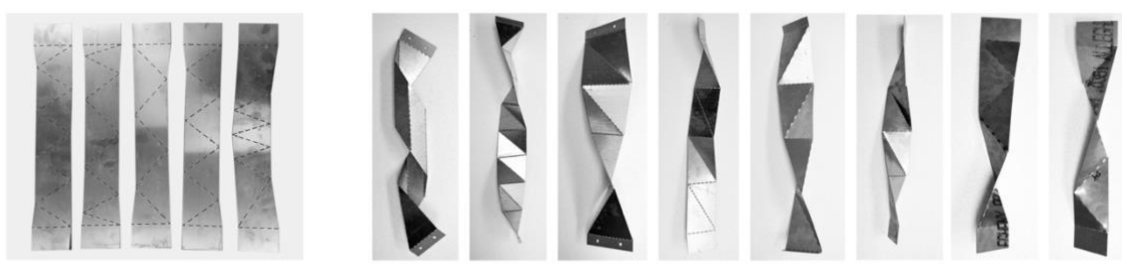

**Figure 7-17. Fold line patterns for sheet metal twist folding.**

Various manual tests with different numbers of subdivisions showed that for the chosen size of the final module and final 180-degree rotation, a subdivision with eight fold lines (i.e., six diagonal lines in the center and two straight lines at the two ends of the module) created eight triangles (i.e., six isosceles and two right triangles at the two ends), producing the most evenly folded outcomes. After the manual tests and identification of the best fold pattern, a second series of tests was performed in order to find the required total degree of rotation for the robot that would achieve the intended 180 degrees of rotation of the part by simultaneously compensating for the overall springback in the six fold lines.

Through empirical testing, we identified two factors affecting the robot's operation: the amount of springback in the final folded part at various rotation angles, and the effect of the rotation speed on the folding process, especially the even folding of all fold lines during a single continuous rotary movement. The final folded module with 180 degrees of rotation was derived with a total rotation angle (on Axis 6) of 210 degrees. This resulted in 30 degrees of springback. However, based on the earlier study described in Section 7.2.3, each fold required an additional 9 degrees to compensate for springback, making the total 9 x 8 = 54 degrees.

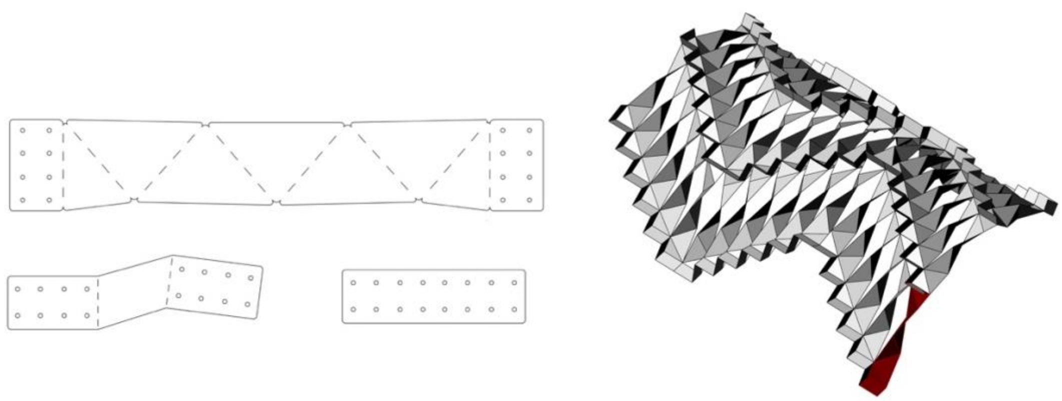

**Figure 7-18. Left: final module with six subdivisions; Right: final structure with folded and assembled units.**

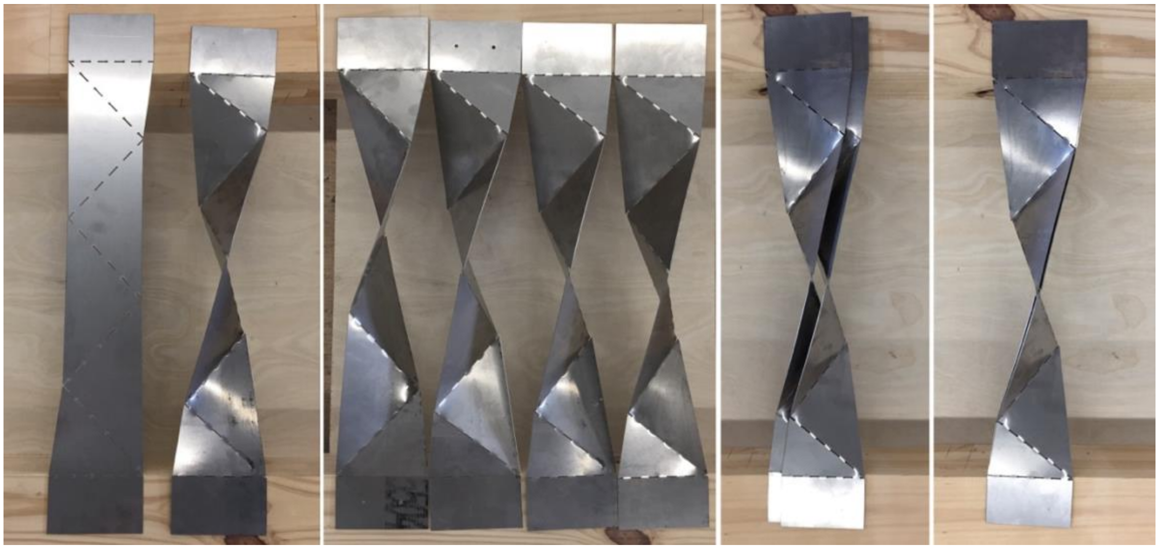

**Figure 7-19. Test models for twist folding.**

Finally, a rotation angle of 210 degrees was identified for Axis 6, which had to be broken into two steps of 90 degrees and 120 degrees. Otherwise, the robot controller wanted to adjust for the shortest path and perform a 150 degree maneuver (360 degrees – 210 degrees) to reach from the start to the end position. The final settings for the geometry, material properties, and robot settings were tested multiple times on modules with identical shapes. The resulting folded parts proved to be identical (see Figure 7-19). After this step, the final fabrication process for the eventual structure was performed and the parts were assembled manually (see Figure 7-20 Figure 7-21).

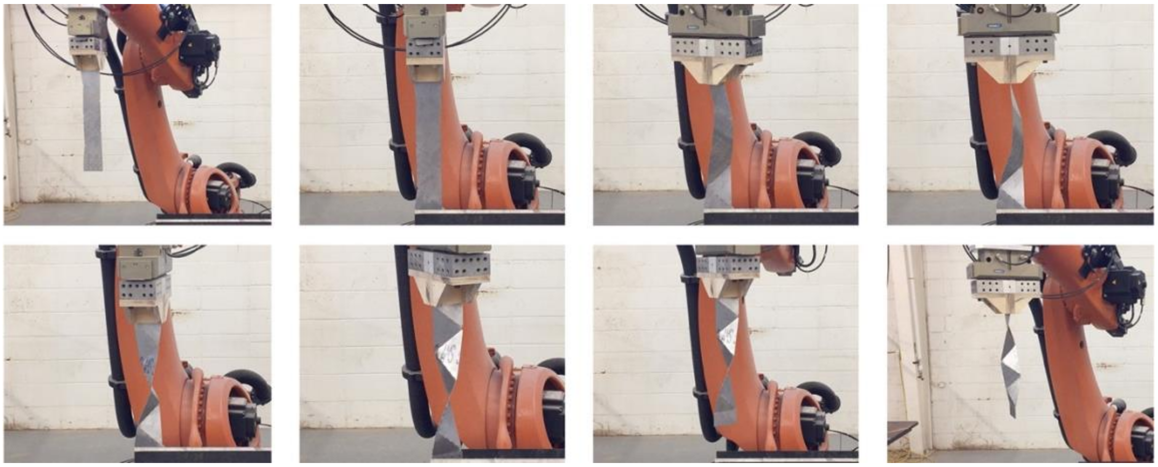

**Figure 7-20. Project 2, the final bending sequence.**

**Figure 7-21. Assembly process for Project 2.**

7.3.2.1 Finite Element Analysis

We performed an empirical investigation to find the most appropriate robot settings for fabrication, given the desired shape and material. However, further analysis is required to identify and quantify the main impact factors on the modeling and fabrication processes. Computer simulation of the formation process using the finite element methods (FEM). This allows the final shape to be predicted, given the forming tools (e.g., dies, molds) and material properties, including the springback effect. The more complex challenge is the inverse problem (i.e., given the desired shape, determining the tool geometry and material properties to use). The first step is to estimate the material's properties, as shown by De Carvalho et al. (De-Carvalho, Valente, and Andrade-Campos 2011) and Chaparro et al. (Chaparro et al. 2008). If the tool geometry is expressed parametrically, optimization methods can be developed for automated parameter identification (Valente et al. 2011, Ponthot and Kleinermann 2006).

Von Misses stress contours from folding done for the present research are shown in Figure 7-22. The contours indicate that the onset of stress occurred at the ends of the perforated lines nearest the robotic gripper. The most intense stresses were concentrated at locations where two perforations intersected, predicting the possibility of tearing at these junctions and the consequent need for more concentrated perforations or circular relief in these locations. At the higher levels of the bend/torque sequence, the FEM predicted a general yielding of the part. The bending was much more localized in the experiments, meaning that we probably did not sufficiently weaken the geometry in the model. The FEM did agree with the experiment to the degree that neither predicted stresses approaching the ultimate strength of the material (and thus separation of the assembly along the bend lines).

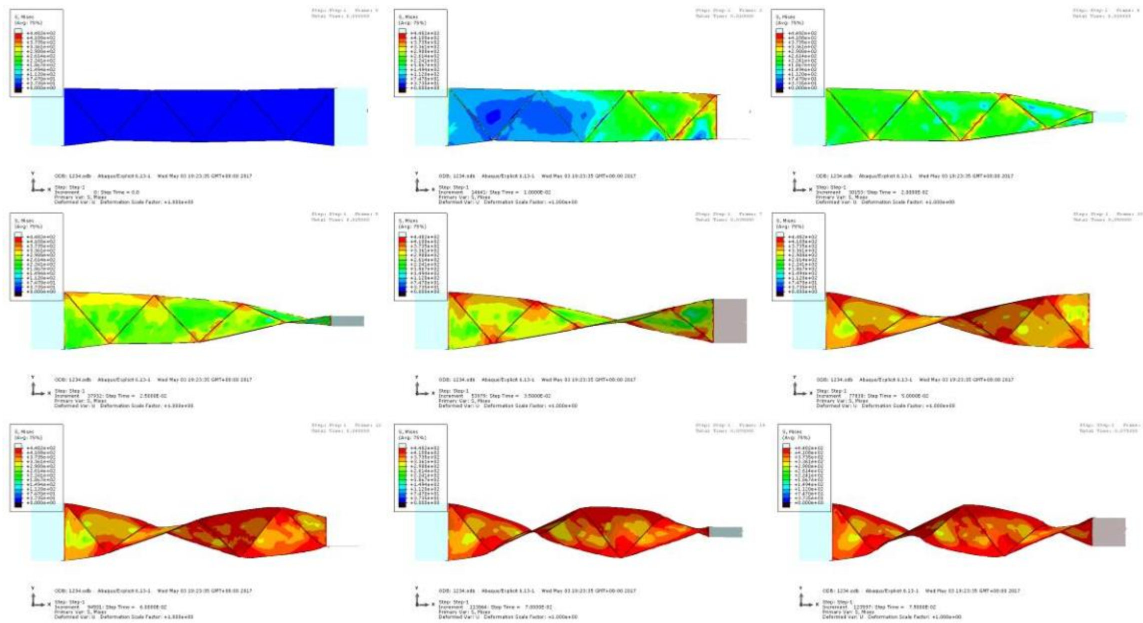

**Figure 7-22. Finite element analysis model of the Project 2 module.**

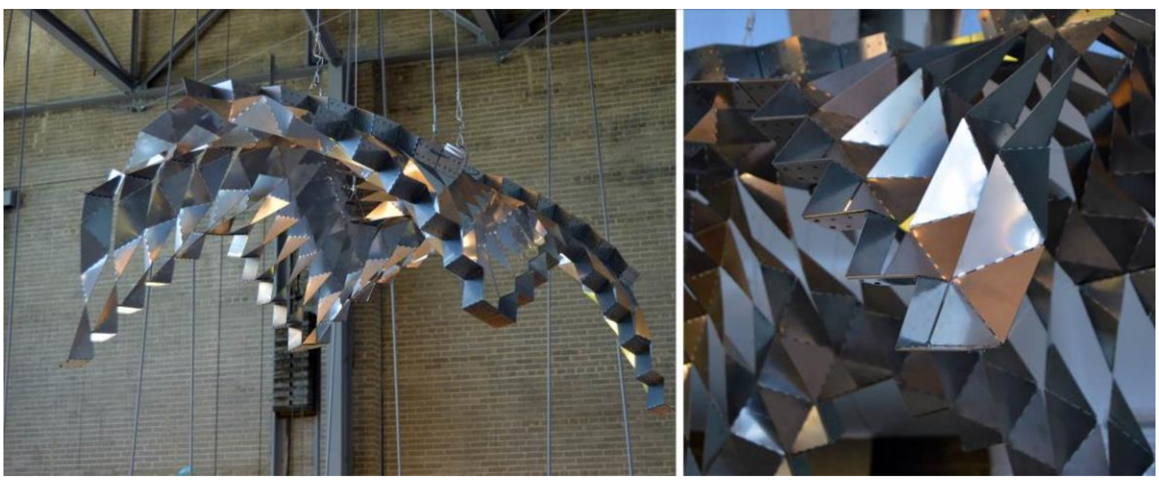

**Figure 7-23. Project 2: the final prototype.**

## 7.4 Discussion and Conclusions

The two robotic metal folding projects performed as a part of this research demonstrated how different approaches to design thinking, reasoning, and problem solving can lead to alternative robotic fabrication solutions. The *tool programming* project proposed a fabrication solution based on a methodical approach that resulted in a reliable and repeatable solution with high levels of precision and accuracy. The fabrication method explored in this project can easily be adopted for the robotic folding of other forms with straight-line folds, after consideration of the main impact factors. These include the geometric unfolding, folding sequence, grasping positions, and springback. However, as discussed above, the path-planning process can be laborious and complex, even after incorporating computational methods.

The *material programming* project explored a novel technique for bypassing individual fold toolpath planning. The twist folding technique was derived based on inductive reasoning and empirical exploration. As a secondary process, through a reverse engineering technique, we tried to identify, analyze, parametrize, and measure the main impact factors for this fabrication process. With this method, we hope to uncover additional ways of applying the technique in future *tool programming* fabrication approaches.

Finally, it should be mentioned that both the deductive engineering and abductive design methods are required for the successful development of creative robotic fabrication techniques. As we move towards an increased application of industrial robotics in prefab and onsite building construction, there is the need for more material-focused robotic fabrication research. Material behaviors, affordances, and limitations can act as deriving factors in the development of new fabrication processes and will inform geometric possibilities. Future extensions of this research will consider data structures required for a design fabrication library (Sharif, Gentry, and Sweet 2016). This library could be used to store parametrized robotic sheet metal fabrication processes, with the dataset enhanced incrementally based on both experimentation and human expert knowledge.

# CHAPTER 8. DISCUSSION AND CONCLUSION

This chapter reviews the problem statement, research question, and hypothesis of this dissertation, and explains the major findings and proposed solution framework. The latter serves as the primary contribution to the greater body of knowledge on this topic. The chapter concludes by addressing the limitations, challenges, and recommendations for future research and development.

## 8.1 The State of Robotic Fabrication in Architectural Practice

Architects have utilized robots for the last several decades to complete mass-customized projects. Robots have exceptional capabilities that enable the fabrication of geometrically complicated components and assembly of complex structures. By using digital fabrication tools, and specifically robots, architects can now create structures that previously were not feasible, to an exceptional level of accuracy, speed, and more importantly, repeatability (Brell-Çokcan and Braumann 2012, McGee and Ponce de Leon 2014, Reinhardt, Saunders, and Burry 2016, Willmann et al. 2018). However, these adopted industrial robots and their control and motion programming tools were all initially intended for engineering-based production practices such as car manufacturing. Current industrial robot control systems force the designer to envision and embed all of the required machining data in the digital model before the fabrication process begins. The designer must predict the material state, tool selection, fixture positioning, and robot motion planning, usually based solely on prior experience. This requirement makes the process of design to fabrication a unidirectional workflow. Consequently, the main challenge for adoption by architects remains finding efficient methods for programming the robots, coordinating their work with the manual processes required, and setting up the system for construction site operations in unique conditions.

Recent research in the field of computational design has identified the technological gap caused by the aforementioned unidirectional design-to-fabrication workflow. In pursuit of a solution, a growing body of research is exploring various human-robot collaboration methods for architectural practices. Researchers have highlighted the potential for humans and machines to act as complementary collaborators in design-to-making processes. Some of these studies have proposed the development of new interfaces for the integration of physical and digital environments, augmenting actuators with sensors to incorporate real-time feedback on the material's state during the production process (Garcia del Castillo Lopez 2019, Munz, Braumann, and Brell-Cokcan 2016, Brugnaro and Hanna 2017, Stumm et al. 2016). However, many of these studies are project-based, targeting the ad hoc needs of a particular robotic application or fabrication process. Consequently, this dissertation investigated a generalizable framework for human-robot collaboration that is rooted in the principles of distributed cognition.

## 8.2 Towards a Human-Robot Collaboration Framework

In Chapter 2, this dissertation investigated human cognition in two different states: the differences in human cognition during manual vs. digital making activities, and engineering vs. design-based digital fabrication practices. Through the lens of distributed cognition, this research explained that human cognition is part of a distributed processing system in which the brain, body, tools, materials, products, and social and material contexts are closely related and interact with one another during an activity (Malafouris 2004, Hollan, Hutchins, and Kirsh 2000, Hutchins 2000). Based on this premise, the main differences between design cognition in traditional craftwork and digital fabrication processes were discussed. As an essential part of the research argument, the role of the tools of production in the formation of a designer's cognitive system was also considered (Clark 2004, McCullough 1998, Norman 1998). For both traditional craftwork and digital fabrication, the tools, materials, and means of production are essential parts of the design process. The main difference is in the way the designer forms the material into the intended final product by using either human or computer-controlled tools. Industrial robots and other computer-controlled machines play an intermediary role in the process, creating a divide between the acts of designing and making.

Additionally, it was explained why engineers and designers have different approaches to their design, experimentation, and production processes. Deductive engineering processes focus on increasing the economy and repeatability of robotic operations, and thus facilitate standardization for the mass production of parts. Conversely, abductive design-based processes prioritize the customization of parts production, with an emphasis on creativity and uniqueness. Designers develop their concepts as a result of experimentation, focusing on the reciprocal exploration of form, testing of material behavior, and determination of robot kinematics.

## 8.3 Research Question and Hypothesis

Chapter 2 argued that a successful human-robot interaction framework should rely on and take advantage of the ability of human intelligence to create tools that extend its core cognitive capabilities. Thus, human-industrial robot collaboration depends on digital technology complementary to the cognitive capabilities of the human brain. An effective collaboration leverages the strengths of the human cognitive system and compensates for its weaknesses by employing the inherent capabilities of robots and computers. Consequently, this dissertation examined the elements required for an interactive human-robot collaboration framework in order to facilitate the creative design of production processes.

## 8.4 Contributions, Challenges, and Limitations

This research addressed the human-robot collaboration framework in two different stages. In the first stage, this dissertation defined a framework for human-robot collaboration that relies on and integrates material and fabrication feedback into the design process. This framework, defined for a bi-directional design and fabrication workflow, has three main components: interactive design, adaptive control, and a design and fabrication library. While different aspects of these components have been studied to various extents by other researchers, this dissertation is the first to define them in an integrated manner.

Next, the requirements for each of these elements were introduced and discussed in detail. *Interactive design*, the first element of this proposed framework, focuses on tools for flexible and intuitive robot control and programming for non-roboticist designers. By using such applications, designers can directly program and simulate industrial robot activity in a parametric modeling environment. Interactive design solutions have been investigated extensively by other researchers, and a few solutions are widely used in the design community. Thus, this dissertation only included a literature review on this topic, and did develop a solution for this issue.

*Adaptive control*, the second piece of the framework, concentrates on solutions for the reciprocal collaboration of humans and robots in order to tackle unpredictable and inaccurate material and environment conditions during the fabrication process. Robot movement and human action are performed based on real-time fabrication data obtained from various sensors. This component assists designers with incorporating uncertainty into the fabrication process. Chapter 3 investigated an adaptive control module by developing a technique for integrating KUKA RSI into the Grasshopper environment. While this dissertation does address this topic, it does not focus on it deeply because many other studies are currently exploring the same problem.

The *library*, the third component, is a knowledge database of design and fabrication methods. This dissertation focused in more detail on the library component of the framework because compared to the first two components, it is the least investigated solution to date. In Chapter 4, a structure for the library was proposed so that the tacit knowledge of makers could be structured, captured, and reused. At its core, the library is a process-centric database where each process is supported by a set of tools, instructions, materials, and geometries required for the transformation of a part into its final form. While this dissertation defines the main entities of the database and relationships among them, it emphasizes that capturing and modeling all of the attributes required for each entity of the library demands an extensive body of research. Consequently, this study focused on demonstrating the generalizability of the library concept through a series of experiments developed for different material systems and with various robotic operations. The main goal of Chapters 5, 6, and 7 was to identify and highlight the commonalities among various fabrication processes and methods so that the requirements of the library could be adequately described.

Chapter 5 focused on the material component of the library. By concentrating on masonry as an example, this part of the dissertation demonstrated the process for identification and classification of a material's generic and specific requirements. It provided a description of the development of a BIM-based industry standard data model for concrete and clay masonry units. The chapter also explained a process for the parametric representation and storage of digital masonry units that would allow for speedy retrieval from a database. Chapter 6 investigated the instruction component of the library, using robotic reinforced masonry fabrication as an example. A new compsite construction technique for reinforced masonry was introduced, and an algorithmic solution for testing the constructability of various complex wall forms was examined. The chapter presented the power of well-defined instruction as a set of rules developed for making infomred design decisions. Finally, Chapter 7 concentrated on the fabrication process aspect of the library, using robotic dieless sheet metal folding as the case study. In addition, the material, geometry, and instruction elements required for the sheet metal folding process were described. The chapter addressed how fabrication methods and processes could be influenced by deductive engineering thinking, as opposed to an abductive design rationale. By presenting the advantages and disadvantages of each approach, it was argued that both engineering and design should be considered necessary and complementary processes.

## 8.5 Recommendations for Future Research and Development

This dissertation defined a roadmap for a human-robot collaboration framework in order to make the design to robotic fabrication processes in architecture a bi-directional workflow. While this dissertation elaborated upon each of the required elements of the framework, further research is needed to comprehensively define and connect all of these requirements. The library component especially requires extensive research on various subtractive, additive, and deformative fabrication processes, as well as the tools, material types, and instructions necessary for each method. A comprehensive library schema can only be defined when all of the attributes for the five major entities are identified.

It should also be considered that while a well-defined structure for a library is needed, it is not sufficient in and of itself. Physical database design, programming, and implementation of the framework and its library are mandatory. The technical software implementation of a comprehensive human-robot collaboration system is a significant endeavor that requires substantial technological infrastructure and resources.

Finally, in addition to the proper definition of the database's data model and appropriate technological implementation, the success of this library relies on its successful adoption and continuous contribution of its users. The data for a functional and successful library should be gathered from experimentation and human expert knowledge inputs and added to the database incrementally over time. Even with a proper data model and database system, it will not be successful if it does not have an active community of users (such as Grasshopper and Dynamo user groups) to add information and draw data from it.

# REFERENCES


"Grasshopper 3D, Algorithmic Modeling for Rhino." https://www.grasshopper3d.com.

"KUKA | prc, Parametric Robot Control." https://www.robotsinarchitecture.org/kuka-prc.

2003. Expert Programming: KUKA System Software (KSS). KUKA Roboter GmbH.

AECMag. 2014. BIM libraries. *AEC Magazine*. Accessed July 01, 2018.

Afsari, Kereshmeh , and Chuck Eastman. 2014. "Categorization of building product models in BIM Content Library portals." Blucher Design Proceedings.

Afsari, Kereshmeh, and Charles Eastman. 2016. "A Comparison of Construction Classification Systems Used for Classifying Building Product Models." Proceedings of the 52nd the Associated Schools of Construction (ASC) Annual International Conference Atlanta, Georgia.

AISC. 2017. The AISC Shapes Database *Steel Construction Manual*. Accessed July 01, 2018.

Albus, Jutta 2018. *Prefabricated Housing: Construction and Design Manual*. Vol. Volume 1: Technologies and Methods. Berlin, Germany: DOM publishers.

Anliker, Franz J. 1988. " Needs for Robots and Advanced Machines at Construction Site." Social aspects of robotics, Proceedings of the 5th ISARC, Tokyo, Japan.

Aomura, Shigeru, and Atsushi Koguchi. 2002. "Optimized bending sequences of sheet metal bending by robot." *Robotics and Computer-Integrated Manufacturing* 18 (1):29-39.

Arnold, A., and B. Wishart. 2008. "Extensible building information model toolset." Google Patents.

ASHRAE. 2016. ASHRAE Standard 90.1-2016: Energy Standard for Buildings Except Low-Rise Residential Buildings. ASHRAE.

ASTM. 2017. Form and Style for ASTM Standards. Accessed July, 01, 2018.

Barak, Ronen, Y-S Jeong, Rafael Sacks, and CM Eastman. 2009. "Unique requirements of building information modeling for cast-in-place reinforced concrete." *Journal of computing in civil engineering* 23 (2):64-74. doi: 10.1061/(ASCE)0887-3801(2009)23:2(64).

Bärtschi, Ralph, Michael Knauss, Tobias Bonwetsch, Fabio Gramazio, and Matthias Kohler. 2010. "Wiggled brick bond." *Advances in Architectural Geometry 2010*:137-147.

Bechthold, Martin. 2010. "The return of the future: a second go at robotic construction." *Architectural Design* 80 (4):116-121.

Benson, Steve. 2014. "Bending Basics: The hows and whys of springback and springforward." *THE FABRICATOR*, July 9, 2014.

Bernal, Marcelo. 2016a. "From Parametric to Meta Modeling in Design." SIGraDi 2016, Argentina, Buenos Aires, 9 - 11 November 2016.

Bernal, Marcelo. 2016b. "Meta-modeling design expertise." PhD, Georgia Institute of Technology.

Bernold, Leonhard E, Frank R Altobelli, and Henry Taylor. 1992. "Computer-controlled brick masonry." *Journal of computing in civil engineering* 6 (2):147-160.

Betti, Giovanni, Saqib Aziz, Andrea Rossi, and Oliver Tessmann. 2018. "Communication Landscapes." Robotic Fabrication in Architecture, Art and Design.

BIA. 2006. Manufacturing of Brick. *Technical Notes on Brick Construction*. Accessed July 01, 2018.

Biggs, David. 2016. "Post-tensioned concrete masonry roof panels: A case study." In *Brick and Block Masonry - Trends, Innovations and Challenges*, edited by C. Modena, F. da Porto and M. R. Valluzzi, 597-602. London: Taylor & Francis Group.

Blikstein, Paulo. 2013. "Digital Fabrication and 'Making'in Education: The Democratization of Invention." *FabLabs: Of Machines, Makers and Inventors*:1-21.

Boerkoel Jr, James C, and Julie A Shah. 2013. "Planning for flexible human-robot co-navigation in dynamic manufacturing environments." International Conference on Human-Robot Interaction (HRI), Proceedings of Human Robot Interaction Pioneers.

Bonwetsch, Tobias. 2012. "Robotic Assembly Processes as a Driver in Architectural Design." *Nexus Network Journal* 14 (3):483-494. doi: 10.1007/s00004-012-0119-3.

Bonwetsch, Tobias. 2015. "Robotically assembled brickwork." TU Darmstadt.

Bonwetsch, Tobias, Ralph Bärtschi, and Matthias Helmreich. 2013. "BrickDesign." Vienna.

Bonwetsch, Tobias, Daniel Kobel, Fabio Gramazio, and Matthias Kohler. 2006. "The informed wall: applying additive digital fabrication techniques on architecture." Proceedings of the 25th Annual Conference of the Association for Computer-Aided Design in Architecture.

Braumann, Johannes, and Sigrid Brell-Cokcan. 2011. "Parametric robot control." *Integrated CAD/CAM for Architectural Design, ACADIA* 2011:242.

Braumann, Johannes, and Sigrid Brell-Cokcan. 2015. "Adaptive Robot Control." Proceedings of the 33rd eCAADe Conference, Vienna, Austria.

Brell-Çokcan, Sigrid, and Johannes Braumann, eds. 2012. *Rob|Arch 2012: Robotic Fabrication in Architecture, Art and Design*: Springer-Verlag Wien.

Brereton, Margot. 2004. "Distributed cognition in engineering design: Negotiating between abstract and material representations." *Design representation*:83-103.

Bruckmann, T, H Mattern, A Spengler, C Reichert, A Malkwitz, and M König. 2016. "Automated Construction of Masonry Buildings using Cable-Driven Parallel Robots." ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction.

Brugnaro, Giulio, Ehsan Baharlou, Lauren Vasey, and Achim Menges. 2016. "Robotic Softness: An Adaptive Robotic Fabrication Process for Woven Structures." Proceedings of the 36th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA) Ann Arbor, MI, 27-29 October, 2016.

Brugnaro, Giulio, Angelo Figliola, and Alexandre Dubor. 2019. "Negotiated Materialization: Design Approaches Integrating Wood Heterogeneity Through Advanced Robotic Fabrication." In *Digital Wood Design*, edited by F. Bianconi and M. Filippucci, 135-158. Springer.

Brugnaro, Giulio, and Sean Hanna. 2017. "Adaptive Robotic Training Methods for Subtractive Manufacturing." Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA), Cambridge, MA.

Cavieres, Andres, and Russell Gentry. 2015. "Masonry Regions: A New Approach for the Representation of Masonry Walls in BIM Applications." Real Time - Proceedings of the 33rd eCAADe Conference, September 2015.

Cavieres, Andres, Russell Gentry, and Tristan Al-Haddad. 2008. "Parametric Design of Masonry Buildings - Embedding Construction Knowledge into Design Models." Proceedings of the 8th International Seminar on Structural Masonry (ISSM), Istanbul, Turkey, 05-07 November 2008.

Cavieres, Andres, Russell Gentry, and Tristan Al-Haddad. 2011. "Knowledge-based parametric tools for concrete masonry walls: Conceptual design and preliminary structural analysis." *Automation in Construction* 20 (6):716-728. doi: 10.1016/j.autcon.2011.01.003.

Celani, Gabriela. 2012. "Digital Fabrication Laboratories: Pedagogy and Impacts on Architectural Education." *Nexus Network Journal* 14 (3):469-482.

Chan, WK, SC Cheung, and TH Tse. 2005. "Fault-Based Testing of Database Application Programs with Conceptual Data Model." Fifth International Conference on Quality Software (QSIC 2005).

Chaparro, BM, Sandrine Thuillier, LF Menezes, Pierre-Yves Manach, and JV Fernandes. 2008. "Material parameters identification: Gradient-based, genetic and hybrid optimization algorithms." *Computational Materials Science* 44 (2):339-346.

Chen, Yan, Rui Peng, and Zhong You. 2015. "Origami of thick panels." *Science* 349 (6246):396-400.

Cheung, Franco KT, Jonathan Rihan, Joseph Tah, David Duce, and Esra Kurul. 2012. "Early stage multi-level cost estimation for schematic BIM models." *Automation in Construction* 27:67-77. doi: 10.1016/j.autcon.2012.05.008.

Clark, Andy. 2001. "Natural-born cyborgs?" In *Cognitive Technology: Instruments of Mind*, 17-24. Springer.

Clark, Andy. 2004. *Natural-born cyborgs: Minds, technologies, and the future of human intelligence*: Oxford University Press.

Correa, David, Athina Papadopoulou, Christophe Guberan, Nynika Jhaveri, Steffen Reichert, Achim Menges, and Skylar Tibbits. 2015. "3D-Printed Wood: Programming Hygroscopic Material Transformations." *3D Printing and Additive Manufacturing* 2 (3):106-116.

Costa, G, and L Madrazo. 2015. "Connecting building component catalogues with BIM models using semantic technologies: an application for precast concrete components." *Automation in Construction* 57:239-248. doi: 10.1016/j.autcon.2015.05.007.

Crowley, Andrew J, and Alastair S Watson. 1997. "Representing engineering information for constructional steelwork." *Computer-Aided Civil and Infrastructure Engineering* 12 (1):69-81. doi: 10.1111/0885-9507.00047.

Dawod, Mohamed, and Sean Hanna. 2019. "BIM-assisted object recognition for the on-site autonomous robotic assembly of discrete structures." *Construction Robotics* 3 (1-4):69-81.

De-Carvalho, R, RAF Valente, and A Andrade-Campos. 2011. "Optimization strategies for non-linear material parameters identification in metal forming problems." *Computers & structures* 89 (1):246-255.

Delimont, Isaac L., Spencer P. Magleby, and Larry L. Howell. 2015. "Evaluating compliant hinge geometries for origami-inspired mechanisms." *Journal of Mechanisms and Robotics* 7 (1):011009.

Dörfler, Kathrin, Florian Rist, and Romana Rust. 2013. "Interlacing: An experimental approach to integrating digital and physical design methods " In *Rob| Arch 2012*, 82-91. Springer.

Dörfler, Kathrin, Timothy Sandy, Markus Giftthaler, Fabio Gramazio, Matthias Kohler, and Jonas Buchli. 2016. "Mobile Robotic Brickwork." In *Robotic Fabrication in Architecture, Art and Design 2016*, 204-217. Springer.

Dritsas, Stylianos. 2015. "A digital design and fabrication library." Proceedings of the Symposium on Simulation for Architecture & Urban Design.

Duflou, Joost R, Dirk Van Oudheusden, J-P Kruth, and Dirk Cattrysse. 1999. "Methods for the sequencing of sheet metal bending operations." *International Journal of Production Research* 37 (14):3185-3202.

Duflou, Joost R., József Váncza, and Richard Aerens. 2005. "Computer aided process planning for sheet metal bending: A state of the art." *Computers in industry* 56 (7):747-771.

dukta-gmbh. "Dukta Flesible Wood." accessed October 10, 2018. https://dukta.com/en/products/.

Eastman, C. M., G. Lee, and R. Sacks. 2002. "Deriving a Product Model from Process Models." Proceedings of the 9th ISPE International Conference on Concurrent Engineering, Cranfield, UK, 27-31 July 2002.

Eastman, Charles. 1999. *Building Product Models: Computer Environments Supporting Design and Construction*: CRC Press.

Eastman, Chuck, Paul Teicholz, Rafael Sacks, and Kathleen Liston. 2011. *A Guide to Building Information Modeling for Owners, Managers, Architects, Engineers, Contractors, and Fabricators*. Hoboken, NJ: John Wiley and Sons,.

Eastman, CM, Y-S Jeong, R Sacks, and I Kaner. 2009. "Exchange model and exchange object concepts for implementation of national BIM standards." *Journal of Computing in Civil Engineering* 24 (1):25-34.

Eastman, CM, Ghang Lee, and Rafael Sacks. 2003. "Development of a Knowledge-Rich CAD System for the North American Precast Concrete Industry." Proceedings of the 2003 Annual Conference of the Association for Computer Aided Design In Architecture, Indianapolis, Indiana, 24-27 October 2003.

Ekholm, Anders, and Lars Häggström. 2011. "Building classification for BIM–Reconsidering the framework." Proceedings of the International Council for Building Research and Innovation in Building Construction (CIB) Congress.

Elashry, Khaled, and Ruairi Glynn. 2014. "An Approach to Automated Construction Using Adaptive Programing." In *Robotic Fabrication in Architecture, Art and Design 2014*, 51-66. Springer.

Epps, Gregory, and Sushant Verma. 2013. "Curved Folding: Design to fabrication process of RoboFold." Shape Modeling International.

Eraut, Michael. 2000. "Non-formal learning and tacit knowledge in professional work." *British Journal of Educational Psychology* 70 (1):113-136.

Feng, C, Y Xiao, A Willette, W McGee, and VR Kamat. 2014. "Towards Autonomous Robotic In-Situ Assembly on Unstructured Construction Sites Using Monocular Vision."

Feng, Chen, Yong Xiao, Aaron Willette, Wes McGee, and Vineet R Kamat. 2015. "Vision guided autonomous robotic assembly and as-built scanning on unstructured construction sites." *Automation in Construction* 59:128–138.

Field, Anne. 2015 "More co-bots work alongside human workers." Cisco, Last Modified Sep. 29, 2015. http://newsroom.cisco.com/feature-content?type=webcontent&articleId=1719855.

Fleming, Katherine, Nicholas Long, and Alex Swindler. 2012. "The Building Component Library: an Online Repository to Facilitate Building Energy Model Creation." Proceedings of the 2012 ACEEE Summer Study on Energy Efficient Buildings, Pacific Grove, California, August 2012.

Fong, Joseph, Francis Pang, and Chris Bloor. 2001. "Converting Relational Database into XML Document." 12th International Workshop on Database and Expert Systems Applications.

Garcia del Castillo Lopez, Jose Luis. 2019. "Enactive Robotics: An Action-State Model for Concurrent Machine Control." Doctor of Design, Graduate School of Design, Harvard University.

García del Castillo y López, Jose Luis. 2019. "Machina. NET: A Library for Programming and Real-Time Control of Industrial Robots." *Journal of Open Research Software* 7:27.

Gentry, Russell, Charles Eastman, Shani Sharif, Tyler Witthuhn, and Jeff Elder. 2014. "Masonry Unit Model Definition: Report 1." Building Information Modeling for Masonry, Phase II Project, Project 1, Pankow Foundation Grant RGA#03-13, March 03, 2014.

Gentry, Russell, Chuck Eastman, and David Biggs. 2013. "A Roadmap for Developing and Deploying Building Information Modeling (BIM) for the Masonry Industry." Georgia Institute of Technology, Digital Building Laboratory, Atlanta, Georgia USA.

Gentry, Russell, Shani Sharif, Andres Cavieres, and David Biggs. 2016. "BIM schema for masonry units and walls." Proceedings of the 16th International Brick and Block Masonry Conference.

Gershenfeld, Neil. 2005. *Fab: the coming revolution on your desktop--from personal computers to personal fabrication*: Basic Books.

Gramazio, Fabio, Matthias Kohler, and Silvan Oesterle. 2010. "Encoding material." *Architectural Design* 80 (4):108-115.

Halim, Steven. 2009. "An integrated White+ Black box approach for designing and tuning stochastic local search algorithms (" PhD, Computer Science, National University of Singapore.

Helm, Volker, Selen Ercan, Fabio Gramazio, and Matthias Kohler. 2012. "Mobile robotic fabrication on construction sites: DimRob." Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on.

Hendry, Arnold W. 1998. *Structural masonry*: Macmillan International Higher Education.

Hoffmann, Hartmut, R Hautmann, and R Petry. 2005. "Studies for the development of a simulation basis for numerically controlled driving of sheet metal." Advanced Materials Research.

Hollan, James, Edwin Hutchins, and David Kirsh. 2000. "Distributed cognition: toward a new foundation for human-computer interaction research." *ACM Transactions on Computer-Human Interaction (TOCHI)* 7 (2):174-196.

Hollan, JD, BB Bederson, and Jonathan Helfman. 1997. "Information Visualization." In *Handbook of Human-Computer Interaction*, edited by M.G. Helander, T.K. Landauer and P.V. Prabhu, 33-48. Elsevier.

Hutchins, Edwin. 1995. *Cognition in the Wild*. Cambridge, MA: MIT press

Hutchins, Edwin. 2000. "Distributed cognition." *Internacional Enciclopedia of the Social and Behavioral Sciences*.

Jaśkowski, Piotr, Anna Sobotka, and Agata Czarnigowska. 2018. "Decision model for planning material supply channels in construction." *Automation in Construction* 90:235-242. doi: 10.1016/j.autcon.2018.02.026.

Jeong, Y. S., C. M. Eastman, R. Sacks, and I. Kaner. 2009. "Benchmark tests for BIM data exchanges of precast concrete." *Automation in construction* 18 (4):469-484. doi: 10.1016/j.autcon.2008.11.001.

Jiménez, P. 2012. "Survey on model-based manipulation planning of deformable objects." *Robotics and computer-integrated manufacturing* 28 (2):154-163.

Johns, Ryan Luke, Axel Kilian, and Nicholas Foley. 2014. "Design Approaches Through Augmented Materiality and Embodied Computation." In *Robotic Fabrication in Architecture, Art and Design 2014*, 319-332. Springer.

Jones, Stephen A., and Harvey M. Bernstein. 2012. The Business Value of BIM in North America, Multi-Year Trend Analysis and User Ratings (2007-2012). *Smart Market Report - McGraw-Hill Construction*. Accessed June 20, 2018.

Kalo, Ammar, and Michael Jake Newsum. 2014. "An investigation of robotic incremental sheet metal forming as a method for prototyping parametric architectural skins." In *Robotic Fabrication in Architecture, Art and Design 2014*, 33-49. Springer.

Kannan, TR, and MS Shunmugam. 2008. "Planner for sheet metal components to obtain optimal bend sequence using a genetic algorithm." *International Journal of Computer Integrated Manufacturing* 21 (7):790-802.

Keller, Charles M, and Janet Dixon Keller. 1993. "Thinking and acting with iron." In *Understanding practice: Perspectives on activity and context*, edited by Seth Chaiklin and Jean Lave, 125-143.

Keller, Charles M, and Janet Dixon Keller. 1996. *Cognition and tool use: The blacksmith at work*: Cambridge University Press.

Kilian, Martin, Simon Flöry, Zhonggui Chen, Niloy J Mitra, Alla Sheffer, and Helmut Pottmann. 2008. "Curved folding." ACM Transactions on Graphics (TOG).

Kinateder, Fred A. 2015. "Contractor Input Project: Final Report." Building Information Modeling for Masonry, Phase II Project, Project 4, April 9, 2015.

Knight, M Dennis. 2012. BIM Object Creation. *ASHRAE Journal, American Society of Heating, Refrigeration and Air Conditioning Engineers, Inc.* 54 (11): 74-75. Accessed July 01, 2018.

Knight, Terry, and Lawrence Sass. 2010. "Looks count: Computing and constructing visually expressive mass customized housing." *Artificial Intelligence for Engineering Design, Analysis and Manufacturing* 24 (03):425-445. doi: 10.1017/S0890060409990126.

Kolarevic, Branko. 2004. *Architecture in the digital age: design and manufacturing*: Taylor & Francis.

Kolko, Jon. 2010. "Abductive thinking and sensemaking: The drivers of design synthesis." *Design Issues* 26 (1):15-28.

Kong, Stephen CW, Heng Li, Yong Liang, Tim Hung, Chimay Anumba, and Zhen Chen. 2005. "Web services enhanced interoperable construction products catalogue." *Automation in construction* 14 (3):343-352. doi: 10.1016/j.autcon.2004.08.008.

Krzywinski, Martin |. 2018. Image Color Summarizer, v0.70. Accessed June 20, 2018.

Lavallee, Justin, Rachel Vroman, and Yair Keshet. 2011. "Automated Folding of Sheet Metal Components with a Six-axis Industrial Robot."

Lee, B. , J. Haymaker, R. Gentry, and Biggs D. 2015. "Developing a Framework for BIM for Masonry through a Systems Modeling and Case Study Approach." 12th North American Masonry Conference, Denver, Colorado, USA.

Lehtinen, Hannu, Esko Salo, and Heikki Aatlo. 1989. "Outlines of Two Masonry Robot Sytems." Proceedings of the 6th International Symposium on Automation and Robotics in Construction, , San Francisco, CA.

Liang, Jianhai, and Ali M Memari. 2011. "Introduction of a panelized brick veneer wall system and its building science evaluation." *Journal of architectural engineering* 17 (1):1-14.

Liao, Xiaoyun, and G. Gary Wang. 2003. "Evolutionary path planning for robot assisted part handling in sheet metal bending." *Robotics and Computer-Integrated Manufacturing* 19 (5):425-430.

Liu, W, and K Tai. 2007. "Optimal design of flat patterns for 3D folded structures by unfolding with topological validation." *Computer-Aided Design* 39 (10):898-913.

Lloret-Fritschi, E, F Scotto, F Gramazio, M Kohler, K Graser, T Wangler, L Reiter, RJ Flatt, and J Mata-Falcón. 2018. "Challenges of real-scale production with smart dynamic casting." RILEM International Conference on Concrete and Digital Fabrication.

Lu, Hui, Hongwei Wang, Yong Xie, and Xiangyu Wang. 2018. "Study on construction material allocation policies: A simulation optimization method." *Automation in Construction* 90:201-212. doi: 10.1016/j.autcon.2018.02.012.

Maidin, Shajahan Bin, Ian Campbell, and Eujin Pei. 2012. "Development of a design feature database to support design for additive manufacturing." *Assembly Automation* 32 (3):235 - 244.

Malafouris, Lambros. 2004. "The cognitive basis of material engagement: where brain, body and culture conflate." In *Rethinking materiality: the engagement of mind with the material world*, edited by Elizabeth DeMarrais, Chris Gosden and Colin Renfrew, 53-61. McDonald Institute for Archaeological Research.

Malafouris, Lambros. 2008. "At the potter's wheel: an argument for material agency." In *Material Agency: Towards a Non-anthropocentric Approach*, edited by Carl Knappett and Lambros Malafouris, 19-36. New York: Springer.

McCullough, Malcolm. 1998. *Abstracting craft: The practiced digital hand*: MIT press.

McGee, Wes, and Monica Ponce de Leon, eds. 2014. *Robotic Fabrication in Architecture, Art and Design 2014*: Springer International Publishing.

Menges, Achim. 2012a. "Material computation: Higher integration in morphogenetic design." *Architectural Design* 82 (2):14-21.

Menges, Achim. 2012b. "Material resourcefulness: activating material information in computational design." *Architectural Design* 82 (2):34-43.

Menges, Achim, and Steffen Reichert. 2012. "Material capacity: embedded responsiveness." *Architectural Design* 82 (2):52-59.

Merriam-Webster. 2020. "'Deduction' vs. 'Induction' vs. 'Abduction'." https://www.merriam-webster.com/words-at-play/deduction-vs-induction-vs-abduction.

Mohanty, Swagatika, Srinivasa Prakash Regalla, and Y. V. Rao. 2015. "Multi-stage and robot assisted incremental sheet metal forming: a review of the state of art and comparison of available technologies." Proceedings of the 2015 Conference on Advances In Robotics.

Monteiro, Ari, Rita Cristina Ferreira, and Eduardo Toledo Santos. 2009. "Representation Paradigms for Masonry Modulation in BIM Tools." *Design Management and Techology (Gestão & Tecnologia de Projetos)* 1 (2):54-75. doi: 10.4237/gtp.v4i2.101.

Moya, Quim, and Oriol Pons. 2014. "Improving the design and production data flow of a complex curvilinear geometric Glass Reinforced Concrete façade." *Automation in Construction* 38:46-58. doi: 10.1016/j.autcon.2013.10.025.

Mueller, Stefanie, Pedro Lopes, and Patrick Baudisch. 2012. "Interactive construction: interactive fabrication of functional mechanical devices." Proceedings of the 25th annual ACM symposium on User interface software and technology.

Munz, Heinrich, Johannes Braumann, and Sigrid Brell-Cokcan. 2016. "Direct Robot Control with mxAutomation: A New Approach to Simple Software Integration of Robots in Production Machinery, Automation Systems, and New Parametric Environments." In *Robotic Fabrication in Architecture, Art and Design 2016*, 440-447. Springer.

Navathe, Shamkant B, and Ramez Elmasri. 2010. *Fundamentals of database systems*. Upper Saddle River, NJ: Pearson Education.

Nawari, Nawari O. 2011. "Masonry BIM and The Structural Domain." Proceedings of the 11th North American Masonry Conference (NAMC), Minneaplois, MN, June 2011.

NCMA. 1997. *Concrete Masonry Shapes and Sizes Manual*: National Concrete Masonry Association

Nersessian, Nancy. 2008. *Creating scientific concepts*: The MIT Press.

Nicholas, Paul, and Gabriella Rossi. 2018. "Modelling a Complex Fabrication System: New design tools for doubly curved metal surfaces fabricated using the English Wheel." In *Proceedings of eCAADe 2018: Computing for a better tomorrow*, 811-820.

Nikolaidis, Stefanos, Przemyslaw Lasota, Gregory Rossano, Carlos Martinez, Thomas Fuhlbrigge, and Julie Shah. 2013. "Human-robot collaboration in manufacturing: Quantitative evaluation of predictable, convergent joint action." Robotics (ISR), 2013 44th International Symposium on.

Nonaka, Ikujiro, and Noboru Konno. 2005. "The Concept of " Ba": Building a Foundation for Knowledge Creation " *Knowledge management: critical perspectives on business and management* 2 (3):53.

Norman, Donald A. 1998. "Being Analog." In *The invisible computer: why good products can fail, the personal computer is so complex, and information appliances are the solution*. MIT press.

Nour, M. 2010. "A Dynamic Open Access Construction Product Data Platform." *Automation in Construction* 19 (4):407-418. doi: 10.1016/j.autcon.2009.11.011.

Novak, Joseph D, and Alberto J Cañas. 2006. "The origins of the concept mapping tool and the continuing evolution of the tool." *Information visualization* 5 (3):175-184.

Novak, Joseph D, and Alberto J Cañas. 2007. "Theoretical origins of concept maps, how to construct them, and uses in education." *Reflecting Education* 3 (1):29-42.

O'Connor, Erin. 2006. "Glassblowing tools: Extending the body towards practical knowledge and informing a social world." *Qualitative sociology* 29 (2):177-193.

OCCS. 2015. The OmniClass Construction Classification System. Accessed October 08, 2015.

Ofluoglu, Salih, Richard Coyne, and John Lee. 2002. "PLA (id): a tool for organising and sharing on-line building product information." *Automation in construction* 11 (5):585-596. doi: 10.1016/S0926-5805(01)00068-1.

Opritescu, Daniel, and Wolfram Volk. 2015. "Automated driving for individualized sheet metal part production—A neural network approach." *Robotics and Computer-Integrated Manufacturing* 35:144-150.

Oxman, Neri. 2012. "Programming matter." *Architectural Design* 82 (2):88-95.

Oxman, Rivka, and Robert Oxman. 2010. "New structuralism: design, engineering and architectural technologies." *Architectural Design* 80 (4):14-23.

Patil, Lalit, Debasish Dutta, and Ram Sriram. 2005. "Ontology-based exchange of product data semantics." *IEEE Transactions on Automation Science and Engineering* 2 (3):213-225. doi: 10.1109/TASE.2005.849087.

Payne, Andrew. 2011. "A five-axis robotic motion controller for designers." Proceedings of the 31st Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA).

Peters, Scott, and Robert Belden. 2014. "SAM, the Robotic Bricklayer." *SMART|Dynamics of Masonry*.

Ponthot, J-P, and J-P Kleinermann. 2006. "A cascade optimization methodology for automatic parameter identification and shape/process optimization in metal forming simulation." *Computer Methods in Applied Mechanics and Engineering* 195 (41):5472-5508.

Ponticel, Patrick. 2013. "Ford sheet-metal-forming technology on journey from lab to application." SAE International. http://articles.sae.org/12385/.

Qattawi, A, A Mayyas, and MA Omar. 2013. "An investigation of graph traversal algorithms in folded sheet metal parts design." *The International Journal of Advanced Manufacturing Technology* 69 (9-12):2237-2246.

Qattawi, A, Ahmad Mayyas, H Thiruvengadam, V Kumar, S Dongri, and M Omar. 2014. "Design considerations of flat patterns analysis techniques when applied for folding 3-D sheet metal geometries." *Journal of intelligent manufacturing* 25 (1):109-128.

Ramakrishnan, Raghu, and Johannes Gehrke. 2000. *Database Management Systems*. 2nd ed: McGraw-Hill College.

Raspall, Felix, Felix Amtsberg, and Stefan Peters. 2014. "Material Feedback in Robotic Production." In *Robotic Fabrication in Architecture, Art and Design 2014*, 333-345. Springer.

Reber, Arthur S. 1989. "Implicit learning and tacit knowledge." *Journal of experimental psychology: General* 118 (3):219.

Reinhardt, Dagmar, Rob Saunders, and Jane Burry, eds. 2016. *Robotic Fabrication in Architecture, Art and Design 2016*: Springer International Publishing.

Rundell, Rick. 2008. BIM and Building Product Manufacturers. *Cadalyst Magazine*. Accessed July 01, 2018.

Sanfilippo, Filippo, Lars Ivar Hatledal, Houxiang Zhang, Massimiliano Fago, and Kristin Y Pettersen. 2015. "Controlling Kuka industrial robots: Flexible communication interface JOpenShowVar." *Robotics & Automation Magazine, IEEE* 22 (4):96-109.

Schön, Donald A. 1992. "Designing as reflective conversation with the materials of a design situation." *Knowledge-Based Systems* 5 (1):3-14.

Schwartz, Thibault. 2013. "HAL." In *Rob| Arch 2012*, 92-101. Springer.

Schwinn, Tobias. 2017. "Landesgartenschau Exhibition Hall." In *Advancing Wood Architecture: A Computational Approach*, edited by Achim Menges, Tobias Schwinn and Oliver David Krieg, 111-123. Routledge.

Sharif, Shani. 2013. "Material Cognition: Designer's Perception Of Material In A Creative Design Process." SIGRADI, Valparaiso, Chile.

Sharif, Shani , and Russell Gentry. 2015a. "BIM for Masonry: Development of BIM Plugins for the Masonry Unit Database." Real Time - Proceedings of the 33rd Education and research in Computer Aided Architectural Design (eCAADe) Conference, Vienna, Austria 16-18 September 2015.

Sharif, Shani, Varun Agrawal, and Larry Sweet. 2017. "Adaptive Industrial Robot Control for Designers." ShoCK - Proceedings of the 35th eCAADe Conference, University of Rome, Rome, Italy.

Sharif, Shani, and Russell Gentry. 2015b. "Design Cognition Shift from Craftsman to Digital Maker." 20th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA 2015), Daegu, Korea.

Sharif, Shani, T Russell Gentry, and Larry M Sweet. 2016. "Human-Robot Collaboration for Creative and Integrated Design and Fabrication Processes." ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction.

Shepherd, Stuart, and Alois Buchstab. 2014. "Kuka robots on-site." In *Robotic Fabrication in Architecture, Art and Design 2014*, 373-380. Springer.

Smith, Ryan E, and John D Quale. 2017. *Offsite architecture: constructing the future*: Taylor & Francis.

Stumm, Sven, Johannes Braumann, Martin von Hilchen, and Sigrid Brell-Cokcan. 2016. "On-Site Robotic Construction Assistance for Assembly Using A-Priori Knowledge and Human-Robot Collaboration." International Conference on Robotics in Alpe-Adria Danube Region.

Tachi, Tomohiro, and Gregory Epps. 2011. "Designing One-DOF mechanisms for architecture by rationalizing curved folding." International Symposium on Algorithmic Design for Architecture and Urban Design (ALGODE-AIJ). Tokyo.

Tibbits, Skylar. 2014. "4D printing: multi-material shape change." *Architectural Design* 84 (1):116-121.

Trubiano, F, J Dessi-Olive, and R Gentry. 2019. "Masonry tectonics: Craft, labor, & structural innovation in architectural education." Structures and Architecture-Bridging the Gap and Crossing Borders: Proceedings of the Fourth International Conference on Structures and Architecture (ICSA 2019), July 24-26, 2019, Lisbon, Portugal.

US-GBC. 2011. *Advanced Energy Modeling for LEED Technical Manual v2.0*: U.S. Green Building Council.

Valdes, Francisco , Andres Cavieres, and Russell Gentry. 2013. "A Process-Centric Approach for Teaching Digital Fabrication." SIGRADI, Valparaiso, Chile.

Valente, Robertt AF, António Andrade-Campos, José F Carvalho, and Paulo S Cruz. 2011. "Parameter identification and shape optimization: An integrated methodology in metal forming and structural applications." *Optimization and Engineering* 12 (1-2):129-152.

von Rosing, Mark, Stephen White, Fred Cummins, and Henk de Man. 2015. "Business Process Model and Notation—BPMN." In *The Complete Business Process Handbook: Body of Knowledge from Process Modeling to BPM*, edited by Mark von RosingAugust-Wilhelm ScheerHenrik von Scheel, 433-457. Boston: Morgan Kaufmann.

Wang, Lihui, Weiming Shen, Helen Xie, Joseph Neelamkavil, and Ajit Pardasani. 2002. "Collaborative conceptual design—state of the art and future trends." *Computer-Aided Design* 34 (13):981-996.

Willmann, Jan , Philippe Block, Marco Hutter, Kendra Byrne, and Tim Schork, eds. 2018. *Robotic Fabrication in Architecture, Art and Design 2018*: Springer, Cham.

Xu, Weiguo, Dan Luo, and Yuan Gao. 2019. "Automatic brick masonry system and its application in on-site construction."

Zhang, Jinyue, and Zhonggui Xing. 2013. "An IFC-based semantic framework to support BIM content libraries." Proceedings of the 30th International Conference of IT in Construction (CIB W78), Beijing, China, October 9-12, 2013.